Custom Installation Notes:
Serial programming and extended IR interface for Arcam Solo Mini

Model covered: Solo Mini
Document Issue 2

Document history:

Issue 1	November 2007	Mark May	First release
Issue 2	January 2008	Matt Neighbour	Added commands specific to USB
Issue 3	January 2008	Matt Neighbour	Changed USB command codes to avoid conflict with Solo Movie

Applicability

Equipment covered
This publication applies to the following inversions of the software and RS232 protocol used by the model:
Solo Mini – protocol version 1.0
Solo Mini – software version 1.3 onwards

Controlling the Arcam Solo Mini via RS232

Introduction
Solo Mini is fitted with an RS232 serial connector that allows remote control from a PC, or similar device. This section of the document describes the protocol for controlling Solo Mini via the RS232 interface.

Conventions

< The remote controller is referred to as the 'RC'.

< All values in this section are hexadecimal values, unless otherwise specified.

Serial cable specification
The cable is wired as a null modem:
Connector 1 Connector 2 Function
2 3 Rx 3— Tx
3 2 Tx —4 Rx
5 5 RS232 ground

Data transfer format

< Transfer rate: 19,200bps.

< 1 start bit, 8 data bits, 1 stop bit, no parity, no flow control.

Command and response formats
Communication between the RC and Solo Mini takes the form of sequences of bytes, with all commands and responses having the same basic format. Communication between the two is full duplex. Solo Mini shall always respond to a received command. Each transmission by the RC shall be at least six bytes in the following format:
<STR> <ZN> <CC> <NB> <Data> <ETR>

< STR (Start transmission): 0x21

< ZN (Zone)

< CC (Command code): the code for the command

< NB (Number of bytes): the number of data items following this item, excluding the ETR

< Data: [Byte(1) - Byte(NB)]

< ETR (End transmission): 0xD

Each response by Solo Mini shall be at least seven bytes in the following format:
<STR> <ZN> <CC> <RC> <NB> <Data> <ETR>

< STR (Start transmission): 0x21

< ZN (Zone)

< CC (Command code)

< RC (Reply code)

< NB (Number of bytes): the number of data items following this item, excluding the ETR

< Data: [Byte(1) - Byte(NB)]

< ETR (End transmission): 0xD

Answer codes
The following answer codes are defined:

Status update 0x00
Command accepted, and complete 0x01
Command accepted, currently processing 0x02
Zone Invalid 0x82
Command not recognised 0x83
Parameter not recognised 0x84
Command invalid at this time 0x85
Invalid data length 0x86

Unless detailed otherwise, the data returned is valid only when the answer code is 'Status update' or 'Command accepted, and complete'. When 'Command accepted, currently processing' is returned, the data returned is the data which is currently being processed.

Status changes
Note that Solo Mini shall transmit messages whenever its status changes. These messages shall be transmitted by Solo Mini at asynchronous intervals.

State changes as a result of other inputs
It is possible that the state of Solo Mini may be changed as a result of user input via the front panel buttons or via the IR remote control. Changes resulting from these inputs shall be relayed to the RC (using the appropriate message type) as they occur. For example, if the user changed the front panel display brightness using the **DISP** button on the remote control, a display message (defined below) would be sent to the RC. A similar action would be taken for all other state changes. Unless stated, all RS232 commands have an associated status message.

Example command and response sequence
As an example, the command to set the front-panel display of the Solo Mini to 'full brightness' (defined below) is as follows:

STR Zone CC DL Data ETR
0x21 0x01 0x0A 0x01 0x04 0x0D
Assuming that the command was accepted by the unit and is being processed, Solo Mini responds to this command with the following sequence:
STR Zone CC AC DL Data ETR
0x21 0x01 0x0A 0x02 0x01 0x04 0x0D

Note: RS232 control is disabled if the iPod option is enabled.

AMX Duet™ Support
Solo Mini is fully compatible with AMX Duet™ Dynamic Device Discovery Protocol (DDDP) The following description of Dynamic Device Discovery comes from the AMX website (**www.amx.com**). Dynamic Device Discovery is part of AMX's Duet™ platform, which combines the proven reliability and power of NetLinx with the extensive capabilities of the Java 2 Micro Edition (J2ME) platform. When integrating a serial or IP device from a manufacturer embedding the Dynamic Device Discovery Protocol (DDDP), Duet recognizes the device and loads the appropriate Duet module, which automatically installs the new device. AMX's NetLinx Master can then find and install the Duet device module either from a library on the master, from AMX's Web site, or from the manufacturer's Web site. Duet also allows for device swapping so that programming changes are not required when devices with DDDP are removed or replaced – a huge benefit for end users. The Duet platform is an extension AMX's InConcert® manufacturer partner program, which was developed to ensure seamless communication between partners' devices and the AMX control system.

Command: "**AMX\r**"
Response: "**AMXB<Device-SDKClass=DiscDevice><Device-Make=ARCAM><Device-Model=solo mini><Device-Revision=x.y.0>\r**"

Where x = rs232 protocol major version number
y = rs232 protocol minor version number

RS232 commands

Status (query only)
Query the current state of Solo Mini.

Byte:	Description:
ZN	0x01 (Zone)
CC	0x00 (Command code)
NB	0x01 (Number of bytes)
DATA	0xF0 = Query

RESPONSE

Byte:	Description:
ZN	0x01 (Zone)
RC	0x00 (Reply Code)
AC	0x01 (Answer code)

NB	0x01 (Number of bytes)
DATA	0x00 = Stand-by state
	0x01 = Power-on
	0x02 = Alarm
	0x03 = Basic menu
	0x04 = N/A
	0x05 = Clock menu
	0x06 = Initialising

Example
Command/response sequence to determine the current state (the unit is currently in standby):
Command: **0x21 0x01 0x00 0x01 0xF0 0x0D**
Response: **0x21 0x01 0x00 0x01 0x01 0x00 0x0D**

Source selection (query only)
Query the source of Solo Mini.

Byte:	Description:
ZN	0x01 (Zone)
CC	0x01 (Command code)
NB	0x01 (Number of bytes)
DATA	0xF0 = Query

RESPONSE

Byte:	Description:
ZN	0x01 (Zone)
RC	0x01 (Reply Code)
AC	0x01 (Answer code)
NB	0x01 (Number of bytes)
DATA	0x00 = N/A
	0x01 = FM
	0x02 = DAB
	0x03 = TAPE
	0x04 = AV
	0x05 = N/A
	0x06 = N/A
	0x07 = AM
	0x08 = GAME
	0x09 = USB
	0x0A = CD
	0x0B = TV
	0x0C = AUX

Example
Command/response sequence to determine the current source (unit is currently set to 'Tape'):
Command: **0x21 0x01 0x01 0x01 0xF0 0x0D**
Response: **0x21 0x01 0x01 0x01 0x01 0x03 0x0D**

Volume
Query/Change the volume level of Solo Mini.

Byte:	Description:
ZN	0x01 (Zone 1)
CC	0x02 (Command code)
NB	0x01 (Number of bytes)
DATA	0x00 = Volume 0
	…
	0x48 = Volume 72
	0xF0 = Query

RESPONSE

Byte:	Description:
ZN	0x01 (Zone)
RC	0x02 (Reply Code)
AC	0x01 (Answer code)
NB	0x01 (Number of bytes)
DATA	0x00 = Volume 0
	…
	0x48 = Volume 72

Example
Command/response sequence to change volume to 36:
Command: **0x21 0x01 0x02 0x01 0x24 0x0D**
Response: **0x21 0x01 0x02 0x01 0x01 0x24 0x0D**

Balance
Query/Change the speaker balance.

Byte:	Description:
ZN	0x01 (Zone)
CC	0x04 (Command code)
NB	0x01 (Number of bytes)
DATA	0x5A = Left +10

...
0x64 = Left/Right 0
...
0x6E = Right +10
0xF0 = Query

RESPONSE

Byte:	Description:
ZN	0x01 (Zone)
RC	0x04 (Reply code)
AC	0x01 (Answer code)
NB	0x01 (Number of bytes)
DATA	0x5A = Left +10
	...
	0x64 = Left/Right 0
	...
	0x6E = Right +10

Example
Command/response sequence to set the output of the left channel, where the new balance is 3dB to the left:
Command: **0x21 0x01 0x04 0x01 0x61 0x0D**
Response: **0x21 0x01 0x04 0x01 0x01 0x61 0x0D**

Bass
Query/Change the bass level.

Byte:	Description:
ZN	0x01 (Zone)
CC	0x05 (Command code)
NB	0x01 (Number of bytes)
DATA	0x5D = -14
	...
	0x63 = -2
	0x64 = 0
	0x65 = +2
	...
	0x6B = +14
	0xF0 = Query

RESPONSE

Byte:	Description:
ZN	0x01 (Zone)
RC	0x05 (Reply code)
AC	0x01 (Answer code)
NB	0x01 (Number of bytes)
DATA	0x5D = -14
	...
	0x63 = -2
	0x64 = 0
	0x65 = +2
	...
	0x6B = +14

Example
Command/response sequence to set the bass, where the new level is +2dB:
Command: **0x21 0x01 0x05 0x01 0x65 0x0D**
Response: **0x21 0x01 0x05 0x01 0x01 0x65 0x0D**

Treble
Query/Change the treble level.

Byte:	Description:
ZN	0x01 (Zone)
CC	0x06 (Command code)
NB	0x01 (Number of bytes)
DATA	0x5D = -14
	...
	0x63 = -2
	0x64 = 0
	0x65 = +2
	...
	0x6B = +14
	0xF0 = Query

RESPONSE

Byte:	Description:
ZN	0x01 (Zone)
RC	0x06 (Reply code)
AC	0x01 (Answer code)
NB	0x01 (Number of bytes)
DATA	0x5D = -14
	...
	0x63 = -2
	0x64 = 0
	0x65 = +2
	...
	0x6B = +14

Example
Command/response sequence to set the treble, where the new level is -12dB:
Command: **0x21 0x01 0x06 0x01 0x5E 0x0D**
Response: **0x21 0x01 0x06 0x01 0x01 0x5E 0x0D**

Mute (query only)
Query the mute state.

Byte:	Description:
ZN	0x01 (Zone 1)
CC	0x08 (Command code)
NB	0x01 (Number of bytes)
DATA	0xF0 = Query

RESPONSE

Byte:	Description:
ZN	0x01 (Zone)
RC	0x08 (Reply code)
AC	0x01 (Answer code)
NB	0x01 (Number of bytes)
DATA	0x00 = Mute Off 0x01 = Mute On

Example
Command/response sequence to query the mute state, where the result is that the output is muted:
Command: **0x21 0x01 0x08 0x01 0xF0 0x0D**
Response: **0x21 0x01 0x08 0x01 0x01 0x01 0x0D**

Tune mode/Preset Select
Query the tune mode/current preset,

Byte:	Description:
ZN	0x01 (Zone)
CC	0x13 (Command code)
NB	0x01 (Number of bytes)
DATA	0x01 = Load Preset 1 … 0x0A = Load Preset 10 0xF0 = Query mode/preset

RESPONSE

Byte:	Description:
ZN	0x01 (Zone)
RC	0x13 (Reply Code)
AC	0x01 (Answer code)
NB	0x02 (Number of bytes)
DATA1	0x00 = Tune mode 0x01 = Preset mode
DATA2	Current Preset Number

Example
Command/response sequence to query the current preset mode, where the result is Preset mode with Preset number 5 loaded:
Command: **0x21 0x01 0x13 0x01 0xF0 0x0D**
Response: **0x21 0x01 0x13 0x01 0x02 0x01 0x05 0x0D**

Display brightness
Query/Change the display brightness.

Byte:	Description:
ZN	0x01 (Zone)
CC	0x0A (Command code)
NB	0x01 (Number of bytes)
DATA	0x00 = Display Off … 0x04 = Display full brightness 0xF0 = Query

RESPONSE

Byte:	Description:
ZN	0x01 (Zone)
RC	0x0A (Reply code)
AC	0x01 (Answer code)
NB	0x01 (Number of bytes)
DATA	0x00 = Display Off … 0x04 = Display full brightness

Example
Command/response sequence to set the display brightness, where the result is that the display brightness is at level 1:
Command: **0x21 0x01 0x0A 0x01 0x01 0x0D**

Response: **0x21 0x01 0x0A 0x01 0x01 0x01 0x0D**

Stand-by display brightness

Query/Change the Stand-by display brightness.

Byte:	Description:
ZN	0x01 (Zone)
CC	0x16 (Command code)
NB	0x01 (Number of bytes)
DATA	0x00 = Display Off
	…
	0x04 = Display full brightness
	0xF0 = Query

RESPONSE

Byte:	Description:
ZN	0x01 (Zone)
RC	0x16 (Reply code)
AC	0x01 (Answer code)
NB	0x01 (Number of bytes)
DATA	0x00 = Display Off
	…
	0x04 = Display full brightness

Example

Command/response sequence to set the display brightness, where the result is that the display brightness is at level 1:

Command: **0x21 0x01 0x16 0x01 0x01 0x0D**
Response: **0x21 0x01 0x16 0x01 0x01 0x01 0x0D**

'Snooze' time

Query/Change the 'snooze' time (when applicable).

Byte:	Description:
ZN	0x01 (Zone)
CC	0x29 (Command code)
NB	0x01 (Number of bytes)
DATA	0x00 = Off
	0x01 = 1 minute
	…
	0x5A = 90 minutes
	0xF0 = Query

RESPONSE

Byte:	Description:
ZN	0x01 (Zone)
RC	0x29 (Reply code)
AC	0x01 (Answer code)
NB	0x01 (Number of bytes)
DATA	0x00 = Off
	0x01 = 1 minute
	…
	0x5A = 90 minutes

Example

Command/response sequence to set the snooze time to 90 minutes:

Command: **0x21 0x01 0x29 0x01 0x5A 0x0D**
Response: **0x21 0x01 0x29 0x01 0x01 0x5A 0x0D**

'Sleep' time

Query/Set the 'sleep' time.

Byte:	Description:
ZN	0x01 (Zone)
CC	0x31 (Command code)
NB	0x01 (Number of bytes)
DATA	0x00 = Off
	0x01 = 1 minute
	…
	0x5A = 90 minutes
	0xF0 = Query

RESPONSE

Byte:	Description:
ZN	0x01 (Zone)
RC	0x31 (Reply code)
AC	0x01 (Answer code)
NB	0x01 (Number of bytes)
DATA	0x00 = Off
	0x01 = 1 minute
	…
	0x5A = 90 minutes

Example

Command/response sequence to set the sleep time to 120 minutes:
Command: **0x21 0x01 0x31 0x01 0x78 0x0D**
Response: **0x21 0x01 0x31 0x01 0x01 0x78 0x0D**

Headphones (query only)

Query whether headphones are connected to Solo Mini.

Byte:	Description:
ZN	0x01 (Zone)
CC	0x32 (Command code)
NB	0x01 (Number of bytes)
DATA	0xF0 = Query

RESPONSE

Byte:	Description:
ZN	0x01 (Zone)
RC	0x32 (Reply Code)
AC	0x01 (Answer code)
NB	0x01 (Number of bytes)
DATA	0x00 = No headphones 0x01 = Headphones connected

Example

Command/response sequence to request the headphone status, where the headphones are not connected:
Command: **0x21 0x01 0x32 0x01 0xF0 0x0D**
Response: **0x21 0x01 0x32 0x01 0x01 0x00 0x0D**

Request tuner module type (query only)

Query the tuner module type present in Solo Mini.

Byte:	Description:
ZN	0x01 (Zone)
CC	0x37 (Command code)
NB	0x01 (Number of bytes)
DATA	0xF0 = Query

RESPONSE

Byte:	Description:
ZN	0x01 (Zone)
RC	0x37 (Reply Code)
AC	0x01 (Answer code)
NB	0x01 (Number of bytes)
DATA	0x00 = UK DAB/FM/AM module fitted 0x01 = US AM/FM module fitted 0x02 = Japan 0x03 = EU

Example

Command/response sequence to request the tuner module type, where the type is UK DAB/FM/AM:
Command: **0x21 0x01 0x37 0x01 0xF0 0x0D**
Response: **0x21 0x01 0x37 0x01 0x01 0x00 0x0D**

Radio station (query only)

Query the current radio station label (DAB/RDS)

Byte:	Description:
ZN	0x01 (Zone)
CC	0xDE (Command code)
NB	0x01 (Number of bytes)
DATA	0xF0 = Request the current station label

RESPONSE

Byte:	Description:
ZN	0x01 (Zone)
RC	0xDE (Reply Code)
AC	0x01 (Answer code)
NB	(Number of bytes)
DATA	The service label of the new radio station, in 8-bit ASCII characters

Example

Command/response sequence to request station label, where the new station is called “DAB STATION 2”:
Command: **0x21 0x01 0xDE 0x01 0xF0 0x0D**
Response: **0x21 0x01 0xDE 0x01 0x0D 0x44 0x41 0x42 0x20 0x53 0x54 0x41 0x54 0x49 0x4F 0x4E 0x20 0x32 0x0D**

Radio station information

Query information on the current radio station.

Byte:	Description:
ZN	0x01 (Zone)
CC	0xDF (Command code)
NB	0x02 (Number of bytes)
D1	0xF0 = Request station information
D2	If the current source is FM/AM: 0x00 = Request the station frequency 0x01 = Request the station signal strength If the current source is DAB: 0x01 = Request the station signal strength 0x02 = Request the station MPEG mode 0x04 = Request the station data rate

RESPONSE

Byte:	Description:
RC	0xDF (Reply Code)
AC	0x01 (Answer code)
NB	0x03 (Number of bytes)
D1	0x00 = Message contains station freq. 0x01 = Message contains signal strength 0x02 = Message contains MPEG mode 0x04 = Message contains data rate
D2	Response to the AM frequency request: MSB of current frequency Response to the FM frequency request: MHz of current frequency Response to the DAB MPEG mode request: 0x00 = Stereo 0x01 = Joint stereo 0x02 = Dual mono 0x03 = Mono Response to the DAB data rate request: 0x00 - 0xC0 (0 - 192kb/s) = data rate Response to the signal strength request: 0x00 - 0x10 = signal strength
D3	Response to the AM frequency request: LSB of current frequency Response to the FM frequency request: kHz/10 of current frequency Response to the DAB MPEG mode request: 0x00 Response to the DAB data rate request: 0x00 Response to the signal strength request: 0x00

Example

Command/response sequence to request the station frequency, where the source is AM and the frequency is 1089kHz:

Command: **0x21 0x01 0xDF 0x02 0xF0 0x00 0x0D**

Response: **0x21 0x01 0xDF 0x01 0x02 0x04 0x41 0x0D**

Example

Command/response sequence to request the station frequency, where the source is FM and the frequency is 105.2MHz:

Command: **0x21 0x01 0xDF 0x02 0xF0 0x00 0x0D**

Response: **0x21 0x01 0xDF 0x01 0x02 0x69 0x32 0x0D**

Example

Command/response sequence to request the signal strength, where the current signal strength is 10:

Command: **0x21 0x01 0xDF 0x02 0xF0 0x01 0x0D**

Response: **0x21 0x01 0xDF 0x01 0x02 0x0A 0x00 0x0D**

Radio programme type (query only)

Query the current station programme type (DAB/RDS only).

Byte:	Description:
ZN	0x01 (Zone)
CC	0xE6 (Command code)
NB	0x01 (Number of bytes)
DATA	0xF0 = Request programme type

RESPONSE

Byte:	Description:
ZN	0x01 (Zone)
RC	0xE6 (Reply Code)
AC	0x01 (Answer code)
NB	(Number of bytes)
DATA	The programme type of the selected station, in 8 bit ASCII characters

Example

Request RDS/DLS information (query only)

Request RDS/DLS (FM/DAB) information from the current radio station.

Byte:	Description:
ZN	0x01 (Zone)
CC	0xE7 (Command code)

NB	0x01 (Number of bytes)
DATA	0xF0 = Request RDS/DLS information

RESPONSE

Byte:	Description:
ZN	0x01 (Zone)
RC	0xE7 (Reply Code)
AC	0x01 (Answer code)
NB	(Number of bytes <= 128)
DATA	The RDS/DLS information of the selected station, in 8 bit ASCII characters.

Example

Command/response sequence to request the RDS/DLS information, where the information is “Playing your favourite music”:

Command: **0x21 0x01 0xE7 0x01 0xF0 0x0D**

Response: **0x21 0x01 0xE7 0x01 0x1C 0x50 0x6c 0x61 0x79 0x69 0x6E 0x67 0x20 0x79 0x6F 0x75 0x72 0x20 0x66 0x61 0x76 0x6F 0x75 0x72 0x69 0x74 0x65 0x20 0x6D 0x75 0x73 0x69 0x63 0x0D**

Virtual remote

Sends an RC5 system/command code combination via RS232. Solo Mini will respond as if the command was received over IR.

Byte:	Description:
ZN	0x01 (Zone 1)
CC	0xE9 (Command code)
NB	0x02 (Number of bytes)
DATA1	System code
DATA2	Command code

RESPONSE

Byte:	Description:
ZN	0x01 (Zone)
RC	0xE9 (Reply Code)
AC	0x01 (Answer code)
NB	0x02 (Number of bytes)
DATA1	System Code
DATA2	Command Code

Example

Command/response sequence to set decode mode to PLII Movie using the relevant IR code (16-33): (See IR section/document for details)

Command: **0x21 0x01 0xE9 0x02 0x10 0x21 0x0D**

Response: **0x21 0x01 0xE9 0x02 0x02 0x10 0x21 0x0D**

CD Playback status

Query the current CD playback state.

Byte:	Description:
ZN	0x01 (Zone)
CC	0xEC (Command code)
NB	0x01 (Number of bytes)
DATA	0xF0 = Request playback state

RESPONSE

Byte:	Description:
ZN	0x01 (Zone)
RC	0xEC (Reply Code)
AC	0x01 (Answer code)
NB	0x01 (Number of bytes)
DATA	0x01 = Loading 0x02 = Playing 0x03 = Stopped 0x04 = Scanning Back 0x05 = Scanning Forward 0x08 = Tray Open/Empty 0x09 = Paused 0x0D = Track Skipping

Example

Command/response sequence to request the current CD playback, where the tray is currently open:

Command: **0x21 0x01 0xEC 0x01 0xF0 0x0D**

Response: **0x21 0x01 0xEC 0x01 0x01 0x08 0x0D**

CD Play mode

Query the current playback mode.

Byte:	Description:
ZN	0x01 (Zone)
CC	0x54 (Command code)
NB	0x01 (Number of bytes)
DATA	0xF0 = Query

RESPONSE

Byte:	Description:
ZN	0x01 (Zone)
RC	0x54 (Reply Code)
AC	0x01 (Answer code)
NB	0x01 (Number of bytes)
DATA	0x00 = Playmode Off
	BITS 1~0
	00 = Repeat Off
	01 = Repeat Single
	11 = Repeat All
	BIT 2
	0 = Shuffle Off
	1 = Shuffle On
	BIT 3
	0 = Program Off
	1 = Program

Example
Command/response sequence to set the current disc play mode, where the mode is repeat all:
Command: **0x21 0x01 0x54 0x01 0x03 0x0D**
Response: **0x21 0x01 0x54 0x01 0x01 0x03 0x0D**

USB Playback status
Query the current USB playback state.

Byte:	Description:
ZN	0x01 (Zone)
CC	0x58 (Command code)
NB	0x01 (Number of bytes)
DATA	0xF0 = Request playback state

RESPONSE

Byte:	Description:
ZN	0x01 (Zone)
RC	0x58 (Reply Code)
AC	0x01 (Answer code)
NB	0x01 (Number of bytes)
DATA	0x01 = Initialising
	0x02 = Playing
	0x03 = Stopped
	0x04 = Scanning Back
	0x05 = Scanning Forward
	0x08 = No device
	0x09 = Paused
	0x10 = Invalid file
	0x11 = No valid files present
	0x0D = Track Skipping

Example
Command/response sequence to request the current USB playback, where the tray is currently open:
Command: **0x21 0x01 0x58 0x01 0xF0 0x0D**
Response: **0x21 0x01 0x58 0x01 0x01 0x08 0x0D**

USB Play mode
Query the current USB playback mode.

Byte:	Description:
ZN	0x01 (Zone)
CC	0x59 (Command code)
NB	0x01 (Number of bytes)
DATA	0xF0 = Query

RESPONSE

Byte:	Description:
ZN	0x01 (Zone)
RC	0x59 (Reply Code)
AC	0x01 (Answer code)
NB	0x01 (Number of bytes)
DATA	0x00 = Repeat/Shuffle Off
	0x01 = Repeat Track
	0x02 = Repeat Folder
	0x03 = Repeat All
	0x04 = Shuffle All
	0x05 = Repeat Folder Shuffle
	0x06 = Repeat All Shuffle

Example
Command/response sequence to set the current USB play mode, where the mode is repeat all:
Command: **0x21 0x01 0x59 0x01 0x03 0x0D**
Response: **0x21 0x01 0x59 0x01 0x01 0x03 0x0D**

Total CD/USB track time (query only)
Query total track time of the current track.

Byte:	Description:

ZN	0x01 (Zone 1)
CC	0x55 (Command code)
NB	0x01 (Number of bytes)
DATA	0xF0 = Query

RESPONSE

Byte:	Description:
ZN	0x01 (Zone)
RC	0x55 (Reply Code)
AC	0x01 (Answer code)
NB	0x03 (Number of bytes)
DATA1	Hours
DATA2	Minutes
DATA3	Seconds

Example
Total track time 0h 14m 48s:
COMMAND: **0x21 0x01 0x55 0x01 0xF0 0x0D**
Response: **0x21 0x01 0x55 0x01 0x03 0x00 0x0E 0x2C 0x0D**

Total CD/USB playback time (query only)
Query total time of current disc/folder/prog?

Byte:	Description:
ZN	0x01 (Zone 1)
CC	0x56 (Command code)
NB	0x01 (Number of bytes)
DATA	0xF0 = Query

RESPONSE

Byte:	Description:
ZN	0x01 (Zone)
RC	0x56 (Reply Code)
AC	0x01 (Answer code)
NB	0x03 (Number of bytes)
DATA1	Hours
DATA2	Minutes
DATA3	Seconds

Example
Current disc time 0h 74m 02s
COMMAND: **0x21 0x01 0x56 0x01 0xF0 0x0D**
Response: **0x21 0x01 0x56 0x01 0x03 0x00 0x4A 0x02 0x00**

Preset save
Saves a radio station to a specific preset number

Byte:	Description:
ZN	0x01 (Zone 1)
CC	0xDC (Command code)
NB	0x01 (Number of bytes)
DATA	0x01 … 0x0A = Preset number to use

Example
Command/response sequence to save the currently selected radio station to preset number 5
COMMAND: **0x21 0x01 0xDC 0x01 0x05 0x0D**
Response: **0x21 0x01 0xDC 0x01 0x02 0x01 0x05 0x0D**

CD/USB Track number information
Query current track info

Byte:	Description:
ZN	0x01 (Zone 1)
CC	0xE8 (Command code)
NB	0x01 (Number of bytes)
DATA	0xF0 = Query

RESPONSE

Byte:	Description:
ZN	0x01 (Zone)
RC	0xE8 (Reply Code)
AC	0x01 (Answer code)
NB	0x06 (Number of bytes)
DATA1	Current Folder (if applicable)
DATA2	Total Folder (if applicable)
DATA3	MSB of Current Track number
DATA4	LSB of Current Track number
DATA5	MSB of Total Tracks
DATA6	LSB of Total Tracks

Example

Current track is 23/32 in folder 2/6
COMMAND: **0x21 0x01 0xE8 0x01 0xF0 0x0D**
Response: **0x21 0x01 0xE8 0x01 0x06 0x02 0x06 0x00 0x17 0x00 0x20 0x0D**

Request USB track name

Request USB track name for file currently playing

Byte:	Description:
ZN	0x01 (Zone)
CC	0x60 (Command code)
NB	0x01 (Number of bytes)
DATA	0xF0 = Request track name information

RESPONSE

Byte:	Description:
ZN	0x01 (Zone)
RC	0x60 (Reply Code)
AC	0x01 (Answer code)
NB	(Number of bytes <= 40)
DATA	The name of the current playing track, in 8 bit ASCII characters.

Example
Command/response sequence to request the track name, where the information is “Hysteria”:
Command: **0x21 0x01 0x60 0x01 0xF0 0x0D**
Response: **0x21 0x01 0x60 0x01 0x08 0x48 0x79 0x73 0x74 0x65 0x72 0x69 0x61 0x0D**

Request USB folder name

Request USB track name for file currently playing

Byte:	Description:
ZN	0x01 (Zone)
CC	0x61 (Command code)
NB	0x01 (Number of bytes)
DATA	0xF0 = Request folder name information

RESPONSE

Byte:	Description:
ZN	0x01 (Zone)
RC	0x61 (Reply Code)
AC	0x01 (Answer code)
NB	(Number of bytes <= 12)
DATA	The name of the current playing folder, in 8 bit ASCII characters.

Example
Command/response sequence to request the folder name, where the information is “Muse”:
Command: **0x21 0x01 0x61 0x01 0xF0 0x0D**
Response: **0x21 0x01 0x61 0x01 0x04 0x4D 0x75 0x73 0x65 0x0D**

CD/USB Playback time

Query current track playback time,

Byte:	Description:
ZN	0x01 (Zone 1)
CC	0xEB (Command code)
NB	0x01 (Number of bytes)
DATA	0xF0 = Query

RESPONSE

Byte:	Description:
ZN	0x01 (Zone)
RC	0xEB (Reply Code)
AC	0x01 (Answer code)
NB	0x03 (Number of bytes)
DATA1	Hours
DATA2	Minutes (0~59)
DATA3	Seconds (0~59)

Example
Current track at 34m 22s:
COMMAND: **0x21 0x01 0xEB 0x01 0xF0 0x0D**
Response: **0x21 0x01 0xEB 0x01 0x03 0x00 0x22 0x16 0x0D**

CD Disc type

Query current disc type

Byte:	Description:
ZN	0x01 (Zone 1)
CC	0xED (Command code)
NB	0x01 (Number of bytes)
DATA	0xF0 = Query

RESPONSE

Byte:	Description:
ZN	0x01 (Zone)
RC	0xED (Reply Code)
AC	0x01 (Answer code)
NB	0x01 (Number of bytes)
DATA	0x02 = Audio CD

0x08 = CD-ROM
0x20 = Unknown
0x21 = No Disc

Example
Audio CD is loaded:
COMMAND: **0x21 0x01 0xED 0x01 0xF0 0x0D**
Response: **0x21 0x01 0xED 0x01 0x01 0x02 0x0D**

Time command
Gets/sets the Real-Time Clock (RTC)

(Query/set only, no status message is generated for this command)

Byte:	Description:
ZN	0x01
CC	0xF8 (Command code)
NB	0x04 (Number of bytes)
DATA	0xF0 = Query
	or
DATA1	Day
	0x01 = Monday … 0x07 = Sunday
DATA2	Hours
	0x00 = 00:00 (12:00am)
	0x01 = 01:00 (01:00am)
	…
	0x16 = 22:00 (10:00pm)
	0x17 = 23:00 (11:00pm))
DATA3	Minutes
	0x00 = 0 minutes
	…
	0x3B = 59 minutes
DATA4	Seconds
	0x00 = 0 seconds
	…
	0x3B = 59 seconds

Example
Command/response sequence to set clock to 17:30 Friday:
Command: **0x21 0x01 0xF8 0x04 0x05 0x11 0x1E 0x00 0x0D**
Response: **0x21 0x01 0xF8 0x01 0x05 0x11 0x1E 0x00 0x0D**

RS232 Protocol Version
Query RS232 protocol version number

Byte:	Description:
ZN	0x01 (Zone 1)
CC	0xF1 (Command code)
NB	0x01 (Number of bytes)
DATA	0xF0 = Query

RESPONSE

Byte:	Description:
ZN	0x01 (Zone)
RC	0xF1 (Reply Code)
AC	0x01 (Answer code)
NB	0x02 (Number of bytes)
DATA1	Major Version
DATA2	Minor Version

Example
Query rs232 protocol version (version 1.2 is returned):
COMMAND: **0x21 0x01 0xF1 0x01 0xF0 0x0D**
Response: **0x21 0x01 0xF1 0x01 0x02 0x01 0x02 0x0D**

Software Version
Query current software version number

Byte:	Description:
ZN	0x01 (Zone 1)
CC	0xF2 (Command code)
NB	0x01 (Number of bytes)
DATA	0xF0 = Query

RESPONSE

Byte:	Description:
ZN	0x01 (Zone)
RC	0xF2 (Reply Code)

AC	0x01 (Answer code)
NB	0x02 (Number of bytes)
DATA1	Major Version
DATA2	Minor Version

Example
Query current software version (version 1.0 is returned):
COMMAND: **0x21 0x01 0xF1 0x01 0xF0 0x0D**
Response: **0x21 0x01 0xF1 0x01 0x02 0x01 0x00 0x0D**

IR remote commands

The following tables give the IR (infrared) remote commands accepted by the Arcam Solo Mini.

AMP commands

System code 16

Decimal Code Command
16-0 Source TV
16-2 Source AV
16-3 Source DAB
16-5 Source TAPE
16-7 Source CD
16-8 Source FRONT
16-12 Standby
16-13 Mute
16-16 Volume +
16-17 Volume –
16-22 Bass +
16-23 Bass –
16-24 Treble +
16-25 Treble –
16-26 Balance Right
16-27 Balance Left
16-52 Source AM
16-53 Source FM
16-59 Display Brightness
16-80 Navigate Right
16-81 Navigate Left
16-82 Menu
16-85 Navigate Down
16-86 Navigate Up
16-87 OK
16-113 Alarm 1 Toggle
16-114 Alarm 2 Toggle
16-115 Alarm 3 Toggle
16-116 Alarm 4 Toggle
16-117 Snooze
16-118 Sleep
16-119 Mute On
16-120 Mute Off
16-123 Standby Off
16-124 Standby On

TUN commands

System code 17

Decimal Code Command
17-0 Tuner Number 0
17-1 Tuner Number 1
17-2 Tuner Number 2
17-3 Tuner Number 3
17-4 Tuner Number 4
17-5 Tuner Number 5
17-6 Tuner Number 6
17-7 Tuner Number 7
17-8 Tuner Number 8
17-9 Tuner Number 9
17-12 Standby
17-18 Display Brightness
17-32 Preset +
17-33 Preset –
17-37 Mode Preset/Tune
17-38 FM/Mono
17-52 Source DAB/AM
17-53 Source FM
17-60 Tune Mode

17-61 Preset Mode
17-63 Info
17-77 Navigate Right
17-78 Navigate Left
17-85 Navigate Down
17-86 Navigate Up
17-87 Ok
17-91 Direct Preset 1
17-92 Direct Preset 2
17-93 Direct Preset 3
17-94 Direct Preset 4
17-95 Direct Preset 5
17-96 Direct Preset 6
17-97 Direct Preset 7
17-98 Direct Preset 8
17-99 Direct Preset 9
17-100 Direct Preset 10
17-125 Menu

CD commands

System code 20

Decimal Code Command
20-0 CD Number 0
20-1 CD Number 1
20-2 CD Number 2
20-3 CD Number 3
20-4 CD Number 4
20-5 CD Number 5
20-6 CD Number 6
20-7 CD Number 7
20-8 CD Number 8
20-9 CD Number 9
20-12 Standby
20-18 Display Brightness
20-29 CD Repeat
20-32 CD Track Forward
20-33 CD Track Back
20-41 Program
20-45 CD Eject
20-48 CD Pause
20-50 CD Scan Back
20-52 CD Scan Forward
20-53 CD Play
20-54 CD Stop
20-58 Clear
20-59 Repeat A-B
20-64 CD Random
20-67 Menu
20-76 Search
20-77 Display Mode
20-80 Navigate Right
20-81 Navigate Left
20-85 Navigate Down
20-86 Navigate Up
20-87 Ok
20-115 Repeat All
20-116 Repeat Single
20-117 Repeat Off
20-118 Shuffle On
20-119 Shuffle Off
20-123 Standby Off
20-124 Standby On