

Technical Document

BMH05108 eight-electrode body composition analysis module Communication Protocol

Table of contents

ONE, VERSION CHANGE RECORD	3
TWO, AGREEMENT	4
1.HARDWARE FORMAT	4
2. DATA PACKET FORMAT	5
3. REFER TO COMMUNICATION PROCESS	5
4. WEIGHT	6
Weight mode switch 0xA0	6
Weight status reads 0xA1	8
5. IMPEDANCE	9
Impedance mode switch 0xB0	9
Impedance status query 0xB1	10
6. BUILT-IN BODY COMPOSITION ALGORITHM	14
Eight-electrode Body270 algorithm input 0xD0	14
Eight-electrode Body270 algorithm output 0xD0	15
Four-electrode TwoLegs/TwoArms algorithm input 0xD1/0xD2	23
Four-electrode TwoLegs/TwoArms algorithm output 0xD1/0xD2	24
7. PARAMETER READING AND WRITING 0x10	28
Parameter description :	28
8. ISP	29
Get version information 0xE0/ Enter upgrade 0xE1	29
9.APPENDIX 25.....	31
Weighing status	31
Calibration status	31
Impedance measurement status	31
Parameter error type	33
Body type	33

1. Version change history

Version	editor	illustrate	Update time
V0.1	Chen Weijian	initial version	2023/01/31
V 0.2	Chen Weijian	Correction of typo: Get the command number of the version	2023 / 07/26
V 0.3	Chen Weijian	Modify obtaining version and entering upgrade instructions	2023 / 09/23
V0.4	Chen Weijian	Modify the eight-electrode Body270 algorithm input 0xD0 : Add product type Modify the eight-electrode Body270 algorithm to output 0xD0 : Increase segmental muscle rate, increase segmental standard	2024/04/02

2. Protocol

Communication adopts the response method, and all information is in the form of one question and one answer. If no reply is received within 100ms , it will be retransmitted every at least 100ms . The communication interval needs to be greater than 100ms .

The data in the protocol adopts **little-endian mode** , that is, the low bit comes first and the high bit comes last. The parts **highlighted in yellow** in the protocol are only used during production development.

If there is no special instructions, please refer to the following abbreviations:

Slave / slave machine : BMH05108 module

Main / host : APP or other main control module

1) Hardware format

Communication hardware interfaces include serial port, USB and BLE (optional). Whenever the module receives data from the interface, this interface is the current communication hardware interface.

Serial port:

The baud rate is 38400 (default) . The baud rate can be modified through the 0x10 command and can be saved after power failure.

Data bits: 8

Stop bits: 1

No check digit

No flow control

USB :

Product: BMH05108-Body Composition Module

Manufacturer: BEST-HEALTH

Class: CDC

Endpoints: No3->OUT; No1->IN

BLE :

Name: BMH05108

Slave Service UUID : 0xFFB0

Characteristic value UUID : 0xFFB1 Properties : Write Without Response

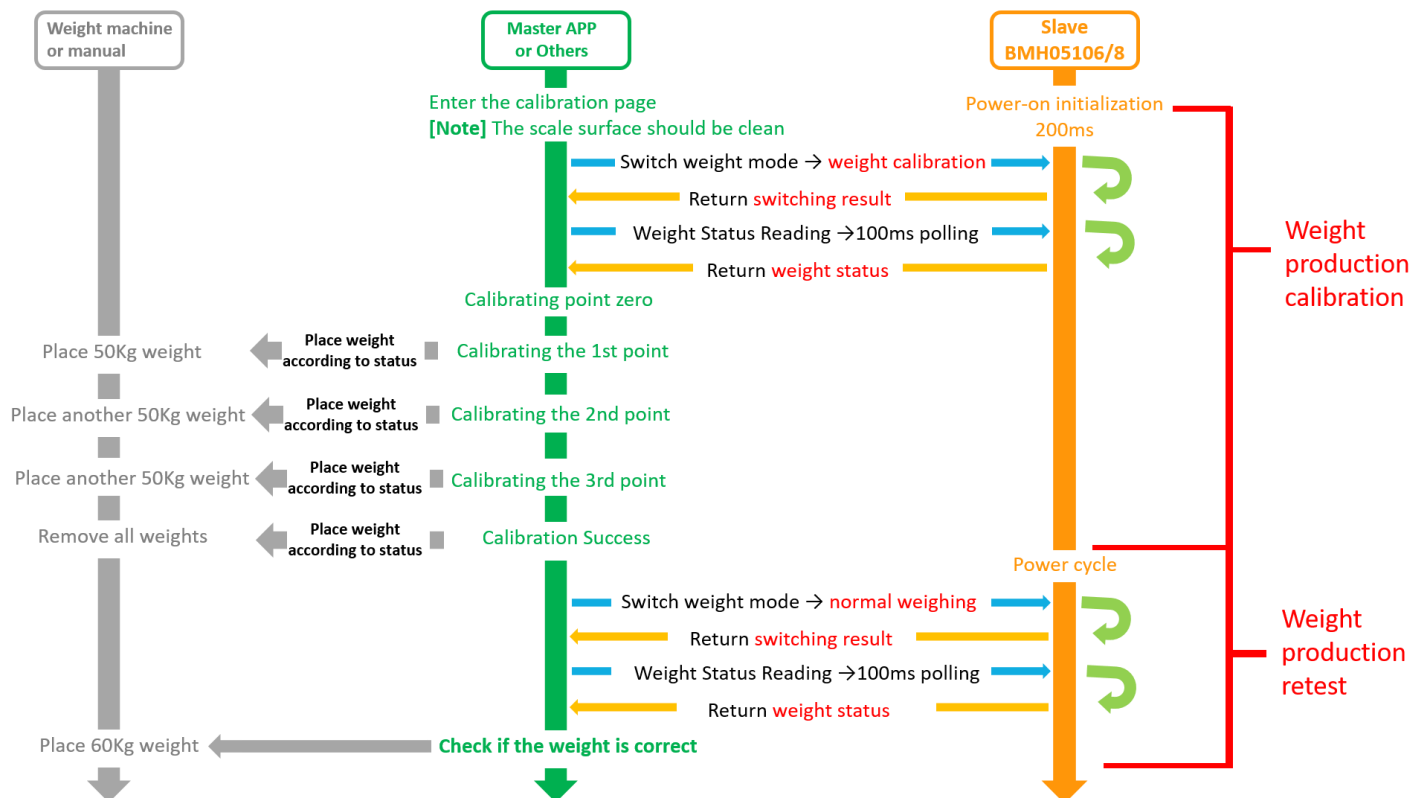
Characteristic value UUID : 0xFFB2 Properties : Read Notify

2) Packet format

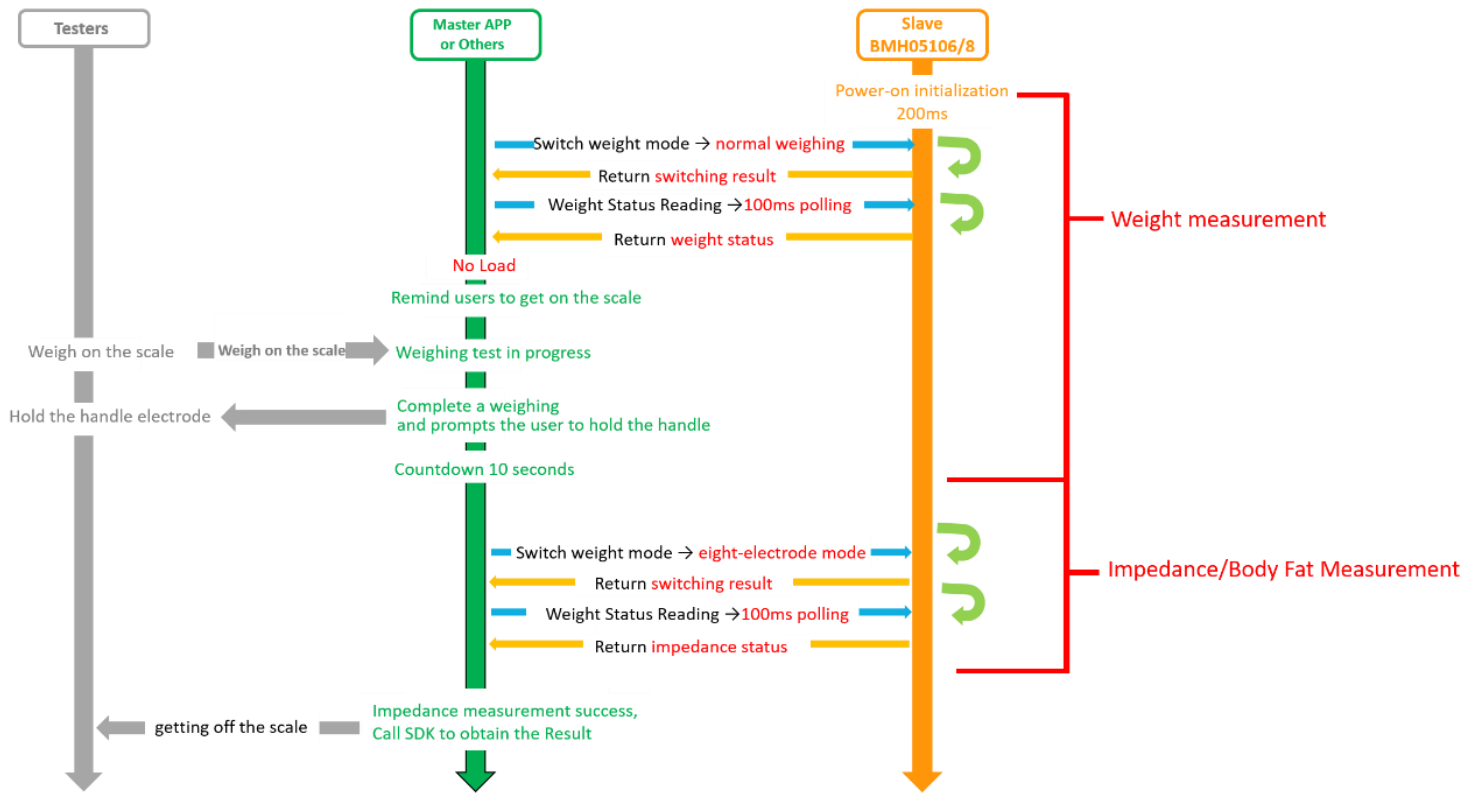
Frame header uint8	Frame length uint8	Order uint8	data (Bytes)	Check Digit uint8
Master- > Slave :0x55	Frame header ~ check bit	0x01~0xFF	... little endian	Checksum = ~ (frame header + command number + ... + one digit before check digit) +1

3) Reference communication process

1. Production calibration



2. Normal use



1. The measurement will be terminated at any time if the user gets off the scale.
2. The 10s countdown is to prevent measurement abnormalities caused by the user not holding the handle. It also allows the user to hold the handle while weighing.
3. Weight and impedance status readings do not affect each other and can be performed simultaneously.

4) weight

Weight mode switch 0xA0

ask

BYTE	type	unit	value	illustrate	
0	Frame	uint8_t	0x55	-	
1	Frame length	uint8_t	0x05		
2	Order	uint8_t	0xA0	0xA0 measurement command	
3	weight mode	uint8_t	0x01-0x03	0x00 invalid value	0x01 Normal weighing
				0x02 weight tare	0x03 Weight Calibration
4	Check Digit	uint8_t	x	Checksum = ~(frame header + command number + ... + one digit	

After the module is powered on, it enters the normal weighing mode by default. You can use this command to switch modes.

Normal weighing : 55 05 A0 01 05

Weight tared : 55 05 A0 02 04

Weight calibration : 55 05 A0 03 03

Weight calibration is only used during production development

answer

BYTE	type	unit	value	illustrate
0	Frame	uint8_t	0xAA	-
1	Frame length	uint8_t	0x05	
2	Order	uint8_t	0xA0	-
3	Switch results	uint8_t	0x01-0x03	0x00 switch OK 0x01 exception. Currently not weighing and cannot tare.
4	Check Digit	uint8_t	x	Checksum = ~ (frame header + command number + ... + one

Normal response: AA 05 A0 00 B1

Abnormal response: AA 05 A0 01 B0 (Tare command can only be performed when the weight is in normal weighing mode)

Weight status reads 0xA1

ask

BYTE	type	unit	value	illustrate
0	Frame header	uint8_t	0x55	-
1	Frame length	uint8_t	0x05	-
2	Order number	uint8_t	0xA1	-
3	data	uint8_t	0x00	reserve
4	Check Digit	uint8_t	x	Checksum = ~ (frame header + command number + ... + one digit before check digit) + 1

Read weight status : 55 05 A1 00 05

answer

BYTE	type	unit	value	illustrate
0	Frame header	uint8_t	0xAA	-
1	Frame length	uint8_t	0x0E	-
2	Order number	uint8_t	0xA1	-
3	weight status	uint8_t	0x00~0xFF	High four digits: weighing status Low four bits: calibration status
4	Peeling sign	uint8_t	0x00~0x01	0x00 : Tare not in progress 0x01 : Tare in progress Affected by mechanisms and sensors, etc. , it usually takes about 1 to 2 seconds to complete.
5-6	stable weight	int16_t	x	The weight result is magnified 10 times, and the unit is catty (=500 grams) For example, 36.5 catty It will be written as 365 , Byte 4 = 0x01 Byte 75= 0x6D
7-8	real time weight	int16_t	x	The weight result is magnified 10 times, and the unit is catty (=500 grams)
9~12	ADC	int32_t	x	ADC value, mainly used for debugging
13	Check Digit	uint8_t	0x00-0xFF	Checksum = ~ (frame header + command number + ... + one digit before check digit) + 1

5) impedance

Impedance mode switch 0xB0

ask

BYTE	type	unit	value	illustrate
0	Frame header	uint8_t	0x55	-
1	Frame length	uint8_t	0x06	
2	Order number	uint8_t	0xB0	
3	Impedance measurement mode	uint8_t	0x00-0x06	0x00 stops the current test
				0x01 Eight-electrode single-frequency fat measurement mode
				0x02 Four-electrode dual-leg single-frequency fat measurement mode
				0x03 Four-electrode bimanual single-frequency fat measurement mode
				0x04 Eight-electrode dual-frequency fat measurement mode (frequency 20kHz & 100kHz)
4	Measurement frequency	uint8_t	0x01-0x09	0x01: 5kHz 0x02: 10kHz 0x03: 20kHz 0x04: 25kHz 0x05: 50kHz 0x06: 100kHz 0x07: 200kHz 0x08: 250kHz 0x09: 500kHz The measurement frequency is valid in single-frequency fat measurement mode and invalid in dual-frequency fat measurement mode.
5	Check Digit	uint8_t	x	Checksum = ~ (frame header + command number +...+ one digit before check digit) +1

Note:

the impedance measurement mode is 0x00 (stop the current test), the impedance history data of all frequencies of the four / eight electrodes will be cleared

the impedance measurement mode is 0x01~0x04 , only the impedance history data of the current frequency in single front mode is cleared.

1. BEST HEALTH eight electrodes 20kHz:55 06 B0 01 03 F1
2. BEST HEALTH eight electrodes 100kHz:55 06 B0 01 06 EE
3. BEST HEALTH four-electrode feet 50kHz:55 06 B0 02 05 EE
4. BEST HEALTH four-electrode two-hand 50kHz:55 06 B0 03 05 ED
5. BEST HEALTH eight electrodes 20kHz & 100kHz:55 06 B0 04 00 F1
6. Stop current test: 55 06 B0 00 00 F5

answer

BYTE	type	unit	value	illustrate
0	Frame header	uint8_t	0xAA	-
1	Frame length	uint8_t	0x05	
2	Order number	uint8_t	0xB0	-
3	Switch results	uint8_t	0x00~0x02	0x00 switch OK 0x01 Working mode error 0x02 Frequency error
4	Check Digit	uint8_t	x	Checksum = ~ (frame header + command number + ... + one digit before check digit) +1

1. Switch OK: AA 05 B0 00 A1
2. Working mode error : AA 05 B0 01 A0
3. Frequency error : AA 05 B0 02 9F

Impedance status query 0xB1

ask

BYTE	type	unit	value	illustrate	
0	Frame header	uint8_t	0x55	-	
1	Frame length	uint8_t	0x05	-	
2	Order number	uint8_t	0xB1	-	
3	response frequency / Response impedance type	uint8_t	x	High four digits (response frequency): 0: Current measurement frequency 1: 5KHz 2: 10KHz 3: 20KHz 4: 25KHz 5: 50KHz 6: 100KHz 7: 200KHz 8: 250KHz 9: 500KHz	Low four bits (response impedance type): 1 : Response to original impedance 2 : Response encryption impedance 3 : Response impedance AD value (retained function)
				(When using body fat algorithms such as BEST HEALTH 's APP/ server, the impedance value needs to be encrypted)	
4	Check Digit	uint8_t	x	Checksum = ~ (frame header + command number + ... + one	

				digit before check digit) +1
--	--	--	--	------------------------------

Request original impedance at current frequency : 55 05 B1 01 F4

Request current frequency encryption impedance : 55 05 B1 02 F3

Request 20kHz raw impedance : 55 05 B1 31 C4

Request 20kHz encryption impedance : 55 05 B1 32 C3

Request 50kHz raw impedance : 55 05 B1 51 A4

Requested 50kHz encryption impedance : 55 05 B1 52 A3

Request 100kHz raw impedance : 55 05 B1 61 94

Request 100kHz encryption impedance : 55 05 B1 62 93

Eight-electrode lipid measurement mode status response

BYTE	type	unit	value	illustrate
0	Frame header	uint8_t	0xAA	-
1	Frame length	uint8_t	0x1B	-
2	Order number	uint8_t	0xB1	-
3	Measurement frequency	uint8_t	0x01~0x09	0x01: 5kHz 0x02: 10kHz 0x03: 20kHz 0x04: 25kHz 0x05: 50kHz 0x06: 100kHz 0x07: 200kHz 0x08: 250kHz 0x09: 500kHz
4	impedance state	uint8_t	0x00~0x06	Impedance measurement status
5	type of data	uint8_t	0x00~0x03	1 : Response to original impedance 2 : Response encryption impedance 3 : Response impedance AD value (retained function) (When using body fat algorithms such as BEST HEALTH's APP/ server, the impedance value needs to be encrypted)
6-9	right hand	uint32_t	x	Resolution 0.1Ω . Example: 4000 is 400.0Ω
10-13	left hand	uint32_t	x	Resolution 0.1Ω . Example: 4000 is 400.0Ω
14-17	trunk	uint32_t	x	Resolution 0.1Ω . Example: 300 is 30.0Ω
18-21	right foot	uint32_t	x	Resolution 0.1Ω . Example: 3000 is 300.0Ω
22-25	left foot	uint32_t	x	Resolution 0.1Ω . Example: 3000 is 300.0Ω
26	Check Digit	uint8_t	0x00-0xFF	Checksum = ~ (frame header + command number + ... + one digit before check digit) +1

Response 20kHz raw impedance : AA 1B B1 03 03 01 A0 0F 00 00 A0 0F 00 00 2C 01 00 00 B8 0B 00 00 B8 0B 00 00 72

Response 20kHz encryption impedance : AA 1B B1 03 03 02 9A 22 93 D0 9A 22 93 D0 44 A2 FB A0
A8 84 F3 31 A8 84 F3 31 23

Response 100kHz raw impedance : AA 1B B1 06 03 01 A0 0F 00 00 A0 0F 00 00 2C 01 00 00 B8 0B 00 00 B8 0B 00 00 6F

Response to 100kHz encryption impedance : AA 1B B1 06 03 02 9A 22 93 D0 9A 22 93 D0 44 A2 FB A0 A8 84 F3 31 A8 84 F3 31 20

Four-electrode lipid measurement mode status response

BYTE	type	unit	value	illustrate
0	Frame header	uint8_t	0xAA	-
1	Frame length	uint8_t	0x0D	-
2	Order number	uint8_t	0xB1	-
3	Measurement frequency	uint8_t	0x01~0x09	0x01: 5KHz 0x02: 10KHz 0x03: 20KHz 0x04: 25KHz 0x05: 50KHz 0x06: 100KHz 0x07: 200KHz 0x08: 250KHz 0x09: 500KHz
4	impedance state	uint8_t	0x00~0x06	Impedance measurement status
5	type of data	uint8_t	0x00~0x03	1 : Response to original impedance 2 : Response encryption impedance 3 : Response impedance AD value (retained function) (When using body fat algorithms such as BEST HEALTH's APP/ server, the impedance value needs to be encrypted)
6-7	phase angle	int16_t	x	The test result is magnified 10 times , for example, 120.5° is written as 1205 That is, Byte 7 = 0xB5; Byte8 = 0x04
8-11	impedance	uint32_t	x	Resolution 1Ω . Example: 480 is 480Ω
12	Check Digit	uint8_t	0x00-0xFF	Checksum = ~ (frame header + command number + ... + one digit before check digit) +1

Response 50kHz raw impedance : AA 0D B1 05 03 01 B5 04 E0 01 00 00 F5

Response to 50kHz encryption impedance : AA 0D B1 05 03 02 B5 04 C6 C6 E4 00 65

6) Built-in body composition algorithm

Eight-electrode Body270 algorithm input 0xD0

ask

BYTE	type	unit	value	illustrate
0	Frame header	uint8_t	0x55	-
1	Frame length	uint8_t	0x27	-
2	Order number	uint8_t	0xD0	-
3	gender	uint8_t	0/1	Gender, range : 0/1(FEMALE/MALE)
4	Product number	uint8_t	x	0 standard algorithm 1~255(customized)
5	height	uint8_t	0x63~0xDC	Height, range : 90~220(cm)
6	age	uint8_t	0x06~0x63	Age, range : 6~99 (years old)
7~8	weight	uint16_t	0x0064~0x07D0	Weight, range : 100~2000 (0.1kg)
9~10	Right hand 20kHz impedance	uint16_t	0x03E8~0x1770	Impedance, range : 1000-6000 (0.1ohm)
11~12	Left hand 20kHz impedance	uint16_t	0x03E8~0x1770	Impedance, range : 1000-6000 (0.1ohm)
13~14	Trunk 20kHz impedance	uint16_t	0x0064~0x03E8	Impedance, range : 100-1000 (0.1ohm)
15~16	Right foot 20kHz impedance	uint16_t	0x03E8~0x1770	Impedance, range : 1000-6000 (0.1ohm)
17~18	Left foot 20kHz impedance	uint16_t	0x03E8~0x1770	Impedance, range : 1000-6000 (0.1ohm)
19~20	Right hand 100kHz impedance	uint16_t	0x03E8~0x1770	Impedance, range : 1000-6000 (0.1ohm)
21~22	Left hand 100kHz impedance	uint16_t	0x03E8~0x1770	Impedance, range : 1000-6000 (0.1ohm)
23~24	Trunk 100kHz impedance	uint16_t	0x0064~0x03E8	Impedance, range : 100-1000 (0.1ohm)
25~26	Right foot 100kHz impedance	uint16_t	0x03E8~0x1770	Impedance, range : 1000-6000 (0.1ohm)
27~28	Left foot 100kHz impedance	uint16_t	0x03E8~0x1770	Impedance, range : 1000-6000 (0.1ohm)
29	Check Digit	uint8_t	Checksum	Checksum = ~ (frame header + command number + ... + one digit before check digit) + 1

Gender :1(MALE)

Product model: 0 (standard algorithm)

Height :0xAC(172cm)

Age :0x17(23 years old)

Weight :0x026F(62.3kg)

right hand

left hand

trunk

right foot

left foot

20kHz 0x0FDB(405.9Ω) 0x10C7(429.5Ω) 0x010C(26.8Ω) 0x0B86(295.0Ω) 0x0BD0(302.4Ω)

100kHz 0x0E06(359.0Ω) 0x0F01(384.1Ω) 0x00DF(22.3Ω) 0x0A35(261.3Ω) 0x0A83(269.1Ω)

55 1E D0 01 00 AC 17 6F 02 DB 0F C7 10 0C 01 86 0B D0 0B 06 0E 01 0F DF 00 35
0A 83 0A 7F

The eight-electrode Body270 algorithm outputs 0xD0

answer

BYTE	type	unit	value	illustrate	
0	Frame header	uint8_t	0xAA	-	
1	Frame length	uint8_t	n+2	-	
2	Order number	uint8_t	0xD0	-	
3	Number of packages	uint8_t	0x00~0xFF	High four digits: total number of packages , for example: total five packages	Example: 0x5n
				The lower four bits: the current number of packets, for example: the nth packet	
4	Error type	uint8_t	0x00~0xFF	Parameter error type	
n	parameter	x	x	When the parameter error type is not ERROR_TYPE_NONE (0x00) , that is, when there is an error in the parameter, the content of this parameter is empty.	
n+1	Check Digit	uint8_t	Checksum	Checksum = ~ (frame header + command number + ... + one digit before check digit) + 1	

The first package: whole body composition parameters

BYTE	type	unit	value	illustrate	
0	Frame header	uint8_t	0xAA	-	
1	Frame length	uint8_t	0x50	-	
2	Order number	uint8_t	0xD0	-	
3	Number of packages	uint8_t	x	High four digits: total number of packages, for example: total five packages	Example: 0x51

				The lower four digits: the current number of packages, for example: the first package	
4	Error type	uint8_t	x	Parameter error type	
5~6	Body weight kg	uint16_t	x	Resolution 0.1kg , example : 6F02 0x026F(623) 62.3	
7~8	Body weight kg (standard Min)	uint16_t	x	Resolution 0.1kg , example : 2902 0x0229(553) 55.3	
9~10	Body weight kg (standard Max)	uint16_t	x	Resolution 0.1kg , example : ED02 0x02ED(749) 74.9	
11~12	Moisture content kg	uint16_t	x	Resolution 0.1kg , example : 6001 0x0160(352) 35.2	
13~14	Moisture content kg (standard Min)	uint16_t	x	Resolution 0.1kg , example : 6D01 0x016D(365) 36.5	
15~16	Moisture content kg (standard Max)	uint16_t	x	Resolution 0.1kg , example : BF01 0x01BF(447) 44.7	
17~18	Body fat mass kg	uint16_t	x	Resolution 0.1kg , example : 8F00 0x008F(143) 14.3	
19~20	Body fat mass kg (standard Min)	uint16_t	x	Resolution 0.1kg , example : 4E00 0x004E(78) 7.8	
21~22	Body fat mass kg (standard Max)	uint16_t	x	Resolution 0.1kg , example : 9C00 0x009C(156) 15.6	
23~24	Protein mass kg	uint16_t	x	Resolution 0.1kg , example : 5F00 0x005F(95) 9.5	
25~26	Protein mass kg (standard Min)	uint16_t	x	Resolution 0.1kg , example : 6200 0x0062(98) 9.8	
27~28	Protein mass kg (standard Max)	uint16_t	x	Resolution 0.1kg , example : 7700 0x0077(119) 11.9	
29~30	Amount of inorganic salt kg	uint16_t	x	Resolution 0.1kg , example : 2100 0x0021(33) 3.3	
31~32	Amount of inorganic salt kg (standard Min)	uint16_t	x	Resolution 0.1kg , example : 2100 0x0021(33) 3.3	
33~34	Inorganic salt content kg (standard Max)	uint16_t	x	Resolution 0.1kg , example : 2900 0x0029(41) 4.1	
35~36	Lean body weight kg	uint16_t	x	Resolution 0.1kg , example : E001 0x01E0(480) 48.0	
37~38	Lean body weight kg (standard Min)	uint16_t	x	Resolution 0.1kg , example : DB01 0x01DB(475) 47.5	
39~40	Lean body weight kg (standard Max)	uint16_t	x	Resolution 0.1kg , example : 5102 0x0251(593) 59.3	
41~42	Muscle mass kg	uint16_t	x	Resolution 0.1kg , example : BF01 0x01BF(447) 44.7	
43~44	Muscle mass kg (standard Min)	uint16_t	x	Resolution 0.1kg , example : D601 0x01D6(470) 47.0	

45~46	Muscle mass kg (standard Max)	uint16_t	x	Resolution 0.1kg , example : 8602 0x0286(646) 64.6
47~48	Bone masskg	uint16_t	x	Resolution 0.1kg , example : 1A00 0x001A(26) 2.6
49~50	Bone mass kg (standard Min)	uint16_t	x	Resolution 0.1kg , example : 1C00 0x001C(28) 2.8
51~52	Bone mass kg (standard Max)	uint16_t	x	Resolution 0.1kg , example : 2300 0x0023(35) 3.5
53~54	Skeletal muscle mass kg	uint16_t	x	Resolution 0.1kg , example : 0801 0x0108(264) 26.4
55~56	Skeletal muscle mass kg (standard Min)	uint16_t	x	Resolution 0.1kg , example : 1601 0x0116(278) 27.8
57~58	Skeletal muscle mass kg (standard Max)	uint16_t	x	Resolution 0.1kg , example : 5401 0x0154(340) 34.0
59~60	Intracellular water volume kg	uint16_t	x	Resolution 0.1kg , example : DC00 0x00DC(220) 22.0
61~62	Intracellular water volume kg (standard Min)	uint16_t	x	Resolution 0.1kg , example : E300 0x00E3(227) 22.7
63~64	Intracellular water volume kg (standard Max)	uint16_t	x	Resolution 0.1kg , example : 1501 0x0115(277) 27.7
65~66	Extracellular water volume kg	uint16_t	x	Resolution 0.1kg , example : 8300 0x0083(131) 13.1
67~68	Extracellular water volume kg (standard Min)	uint16_t	x	Resolution 0.1kg , example : 8B00 0x008B(139) 13.9
69~70	Extracellular water volume kg (standard Max)	uint16_t	x	Resolution 0.1kg , example : AA00 0x00AA(170) 17.0
71~72	Body cell mass kg	uint16_t	x	Resolution 0.1kg , example : 3C01 0x013C(316) 31.6
73~74	Body cell mass kg (standard Min)	uint16_t	x	Resolution 0.1kg , example : 4501 0x0145(325) 32.5
75~76	Body cell mass kg (standard Max)	uint16_t	x	Resolution 0.1kg , example : 8D01 0x018D(397) 39.7
77~78	Subcutaneous fat mass kg	uint16_t	x	Resolution 0.1kg , example : 8100 0x0081(129) 12.9
79	Check Digit	uint8_t	Checksum	Checksum = ~ (frame header + command number +...+ one digit before check digit) +1

Package 2: Segmental fat and muscle information

BYTE		type	unit	value	illustrate
0	Frame header		uint8_t	0xAA	-
1	Frame length		uint8_t	0x2E	-

2	Order number	uint8_t	0xD0	-	
3	Number of packages	uint8_t	x	High four digits: total number of packages, for example: total five packages The lower four digits: the current number of packages, for example: the second package	Example: 0x52
4	Error type	uint8_t	x	Parameter error type	
5~6	Segmental fat mass	Right hand fat mass	uint16_t	x	Resolution 0.1kg , example : 0800 0x0008(8) 0.8
7~8		left hand fat mass	uint16_t	x	Resolution 0.1kg , example : 0900 0x0009(9) 0.9
9~10		Trunk fat mass	uint16_t	x	Resolution 0.1kg , example : 4800 0x0048(72) 7.2
11~12		Right foot fat mass	uint16_t	x	Resolution 0.1kg , example : 1200 0x0012(18) 1.8
13~14		Left foot fat mass	uint16_t	x	Resolution 0.1kg , example : 1300 0x0013(19) 1.9
15~16	Segmental fat rate	Right hand fat percentage	uint16_t	x	Resolution 0.1 %
17~18		Left hand fat percentage	uint16_t	x	Resolution 0.1 %
19~20		trunk fat percentage	uint16_t	x	Resolution 0.1 %
21~22		Right foot fat percentage	uint16_t	x	Resolution 0.1 %
23~24		Left foot fat percentage	uint16_t	x	Resolution 0.1 %
25~26	Segmental fat mass	right hand muscle mass	uint16_t	x	Resolution 0.1kg , example : 1700 0x0017(23) 2.3
27~28		Left hand muscle mass	uint16_t	x	Resolution 0.1kg , example : 1900 0x0019(25) 2.5
29~30		trunk muscle mass	uint16_t	x	Resolution 0.1kg , example : D400 0x00D4(212) 21.2
31~32		Right foot muscle mass	uint16_t	x	Resolution 0.1kg , example : 4B00 0x004B(75) 7.5
33~34		Left foot muscle mass	uint16_t	x	Resolution 0.1kg , example : 4D00 0x004D(77) 7.7
35~36	Segmental muscle ratio	right hand muscle ratio	uint16_t	x	Resolution 0.1 % (supported by Master firmware V1.3 or above)

37~38		Left hand muscle ratio	uint16_t	x	
39~40		trunk muscle ratio	uint16_t	x	
41~42		Right foot muscle ratio	uint16_t	x	
43~44		Left foot muscle ratio	uint16_t	x	
45		Check Digit	uint8_t	Checksum	Checksum = ~ (frame header + command number + ... + one digit before check digit) + 1

The third package: evaluation suggestions

BYTE	type	unit	value	illustrate	
0	Frame header	uint8_t	0xAA	-	
1	Frame length	uint8_t	0x3A	-	
2	Order number	uint8_t	0xD0	-	
3	Number of packages	uint8_t	x	High four digits: total number of packages, for example: total five packages The lower four digits: the current number of packages, for example: the first package	Example: 0x53
4	Error type	uint8_t	x	Parameter error type	
5	body score	uint8_t	x	Resolution 1 , example : 42 0x42(66) 66	
6	physical age	uint8_t	x	Resolution 1 , example : 13 0x13(19) 19	
7	body type	uint8_t	x	Resolution 1 , example : 04 0x04(4) swollen and obese type body type	
8	skeletal muscle mass index	uint8_t	x	Resolution 1 , example : 43 0x43(67) 67	
9	waist to hip ratio	uint8_t	x	Resolution 0.01 , example : 4F 0x4F(79) 0.79	
10	Waist-to-hip ratio (standard Min)	uint8_t	x	Resolution 0.01 , for example : 50 0x50(80) 0.8	
11	Waist-to-hip ratio (standard Max)	uint8_t	x	Resolution 0.01 , example : 5A 0x5A(90) 0.9	
12	Visceral fat level	uint8_t	x	Resolution 1 , example : 05 0x05(5)5	
13	Visceral fat grade (standard Min)	uint8_t	x	Resolution 1 , example : 01 0x01(1)1	
14	Visceral fat grade (standard Max)	uint8_t	x	Resolution 1 , example : 09 0x09(9)9	
15~16	Obesity %	uint16_t	x	Resolution 0.1 , example : BC03 0x03BC(956) 95.6	
17~18	Obesity % (Standard Min)	uint16_t	x	Resolution 1 , example : 5A00 0x005A(90) 90	

19~20	Obesity % (Standard Max)	uint16_t	x	Resolution 1 , example : 6E00 0x006E(110) 110
21~22	Body mass indexBMI kg/m^2	uint16_t	x	Resolution 0.1 , example : D300 0x00D3(211) 21.1
23~24	Body mass index (BMI) kg/m^2 (standard Min)	uint16_t	x	Resolution 0.1 , example : B900 0x00B9(185) 18.5
25~26	Body mass index BMI kg/m^2 (Standard Max)	uint16_t	x	Resolution 0.1 , example : E600 0x00E6(230) 23.0
27~28	Body fat percentage	uint16_t	x	Resolution 0.1 , example : E500 0x00E5(229) 22.9
29~30	Body fat percentage (standard Min)	uint16_t	x	Resolution 0.1 , example : 6400 0x0064(100) 10.0
31~32	Body fat percentage (standard Max)	uint16_t	x	Resolution 0.1 , example : C800 0x00C8(200) 20.0
33~34	Basal metabolism kCal	uint16_t	x	Resolution 1 , example : 7E05 0x057E(1406) 1406
35~36	Basal metabolism kCal (standard Min)	uint16_t	x	Resolution 1 , example : 7705 0x0577(1399) 1399
37~38	Basal metabolism kCal (standard Max)	uint16_t	x	Resolution 1 , example : 5C06 0x065C(1628) 1628
39~40	Recommended intake kCal	uint16_t	x	Resolution 1 , example : 2307 0x0723(1827) 1827
41~42	ideal weight kg	uint16_t	x	Resolution 0.1 , example : 8B02 0x028B(651) 65.1
43~44	Target weight kg	uint16_t	x	Resolution 0.1 , example : 8B02 0x028B(651) 65.1
45~46	Weight control kg	int16_t	x	Resolution 0.1 , example : 1C00 0x001C(28) 2.8
47~48	Muscle control kg	int16_t	x	Resolution 0.1 , example : 4900 0x0049(73) 7.3
49~50	Fat control amount kg	int16_t	x	Resolution 0.1 , example : D3FF 0xFFD3(-45) - 4.5
51~52	Subcutaneous fat rate %	uint16_t	x	Resolution 0.1 , example : CF00 0x00CF(207) 20.7
53~54	Subcutaneous fat rate % (standard Min)	uint16_t	x	Resolution 0.1 , example : 5600 0x0056(86) 8.6
55~56	Subcutaneous fat rate % (Standard Max)	uint16_t	x	Resolution 0.1 , example : A700 0x00A7(167) 16.7
57	Check Digit	uint8_t	Checksum	Checksum = ~ (frame header + command number + ... + one digit before check digit) + 1

The fourth package: exercise consumption

BYTE	type	unit	value	illustrate	
0	Frame header	uint8_t	0xAA	-	
1	Frame length	uint8_t	0x16	-	
2	Order number	uint8_t	0xD0	-	
3	Number of packages	uint8_t	x	High four digits: total number of packages, for example: total five packages	Example: 0x54
				The lower four digits: the current number of packages, for example: the first package	
4	Error type	uint8_t	x	Parameter error type	
5~6	walk	uint16_t	x	Resolution 1 , example : 7C00 0x007C(124) 124kCal/30 minutes	
7~8	golf	uint16_t	x	Resolution 1 , example : 6D00 0x006D(109) 109kCal/30 minutes	
9~10	croquet	uint16_t	x	Resolution 1 , example : 7600 0x0076(118) 118kCal/30 minutes	
11~12	Tennis, cycling, basketball	uint16_t	x	Resolution 1 , example : BA00 0x00BA(186) 186kCal/30 minutes	
13~14	Squash, bouncy ball, taekwondo, fencing	uint16_t	x	Resolution 1 , example : 3701 0x0137(311) 311kCal/30 minutes	
15~16	climb mountains	uint16_t	x	Resolution 1 , example : CB00 0x00CB(203) 203kCal/30 minutes	
17~18	Swimming, aerobics, jogging, football, skipping rope	uint16_t	x	Resolution 1 , example : D900 0x00D9(217) 217kCal/30 minutes	
19~20	Badminton, table tennis	uint16_t	x	Resolution 1 , example : 8C00 0x008C(140) 140kCal/30 minutes	
twenty one	Check Digit	uint8_t	Checksum	Checksum = ~ (frame header + command number +...+ one digit before check digit) +1	

Package 5: Segment Standards

by Master firmware V1.3 or above

BYTE	type	unit	value	illustrate
0	Frame header	uint8_t	0xAA	-
1	Frame length	uint8_t	0x16	-
2	Order number	uint8_t	0xD0	-

3	Number of packages		uint8_t	x	High four digits: total number of packages, for example: total five packages The lower four digits: the current number of packages, for example: the first package	Example: 0x55
4	Error type		uint8_t	x	Parameter error type	
5	Segmental fat standard	right hand	u int8_t	0x00/0x01/0x02	0x00 : low standard 0x01: standard 0x 02: Super standard	
6		left hand	u int8_t	0x00/0x01/0x02		
7		trunk	u int8_t	0x00/0x01/0x02		
8		right foot	u int8_t	0x00/0x01/0x02		
9		left foot	u int8_t	0x00/0x01/0x02		
10	Segmental muscle standards	right hand	u int8_t	0x00/0x01/0x02		
11		left hand	u int8_t	0x00/0x01/0x02		
12		trunk	u int8_t	0x00/0x01/0x02		
13		right foot	u int8_t	0x00/0x01/0x02		
14		left foot	u int8_t	0x00/0x01/0x02		
15 ~ 20	reserve		x	x	reserve	
twenty one	Check Digit		uint8_t	Checksum	Checksum = ~ (frame header + command number + ... + one digit before check digit) +1	

Tx: 55 1E D0 01 00 AC 18 6D 02 ED 0F D8 10 0B 01 8D 0B B8 0B 10 0E 08 0F DD 00 3B 0A 6B 0A 72

Rx: AA 50 D0 51 00 6D 02 29 02 ED 02 62 01 6D 01 BF 01 8A 00 4E 00 9C 00 5F 00 62 00 77 00 22 00 21 00 29 00 E3 01 DB 01 51 02 C1 01 D6 01 86 02 1B 00 1C 00 23 00 0A 01 16 01 54 01 DE 00 E3 00 15 01 83 00 8B 00 AA 00 3E 01 45 01 8D 01 7C 00 90

Rx: AA 2E D0 52 00 08 00 08 00 46 00 12 00 12 00 35 05 35 05 AB 06 22 04 22 04 19 00 18 00 D5 00 4E 00 4C 00 7C 03 59 03 AE 03 E8 0 3 CE 03 33

Rx: AA 3A D0 53 00 43 14 04 44 4F 50 5A 05 01 09 5F 00 5A 00 6E 00 D1 00 B9 00 E6 00 DE 00 64 00 C8 00 85 05 74 05 58 06 2C 07 8B 02 8B 02 1E 00 46 00 D8 FF C7 00 56 00 A7 00 04

Rx: AA 16 D0 54 00 7C 00 6D 00 75 00 BA 00 36 01 CA 00 D9 00 8C 00 9E

Rx: AA 16 D0 55 00 01 01 02 01 01 01 01 01 01 00 00 00 00 00 10

Four-electrode TwoLegs/TwoArms algorithm input 0xD1/0xD2

ask

BYTE	type	unit	value	illustrate
0	Frame header	uint8_t	0x55	-
1	Frame length	uint8_t	0x0C	-
2	Order number	uint8_t	0xD1/0xD2	0xD1: Four-electrode TwoLegs 0xD2: Four-electrode TwoArms
3	gender	uint8_t	0/1	Gender, range : 0/1(FEMALE/MALE)
4	user type	bool	0/1	User type, range : 0/1 (NORMAL/ATHLETE) (reserved function)
5	height	uint8_t	0x63-0xDC	Height, range : 90~220(cm)
6	age	uint8_t	0x06-0x63	Age, range : 6~99 (years old)
7~8	weight	uint16_t	0x0064-0x07D0	Weight, range : 100~2000 (0.1kg)
9~10	50kHz impedance of both feet	uint16_t	0x03E8-0x1770	Impedance, range : 10-1200 (1ohm)
11	Check Digit	uint8_t	Checksum	Checksum = ~ (frame header + command number +...+ one digit before check digit) +1

TwoLegs:

Gender :1(MALE)

User type: 0(NORMAL)

Height :0xAC(172cm)

Age :0x17(23 years old)

Weight :0x026F(62.3kg)

50kHz impedance of both feet : 0x0209 (521Ω)

55 0C D1 01 00 AC 17 6F 02 09 02 8E

TwoArms:

Gender :1(MALE)

User type: 0(NORMAL)

Height :0xAC(172cm)

Age :0x17(23 years old)

Weight :0x026F(62.3kg)

Hands 50kHz impedance : 0x02F6 (758Ω)

55 0C D2 01 00 AC 17 6F 02 F6 02 A0

Four-electrode TwoLegs/TwoArms algorithm output 0xD1/0xD2

BYTE	type	unit	value	illustrate	
0	Frame header	uint8_t	0xAA	-	
1	Frame length	uint8_t	n+2	-	
2	Order number	uint8_t	0xD1	0xD1: Four-electrode TwoLegs 0xD2: Four-electrode TwoArms	
3	Number of packages	uint8_t	0x00~0xFF	High four digits: total number of packages , example: 1	Example: 0x11
				The lower four bits: the current number of packets, for example: 1	
4	Error type	uint8_t	0x00~0xFF	Parameter error type	
n	parameter	x	x	When the parameter error type is not ERROR_TYPE_NONE (0x00) , that is, when there is an error in the parameter, the content of this parameter is empty.	
n+1	Check Digit	uint8_t	Checksum	Checksum = ~ (frame header + command number +...+ one digit before check digit) +1	

BYTE	type	unit	value	illustrate	
0	Frame header	uint8_t	0xAA	-	
1	Frame length	uint8_t	0x56	-	
2	Order number	uint8_t	0xD1	0xD1: Four-electrode TwoLegs 0xD2: Four-electrode TwoArms	
3	Number of packages	uint8_t	0x00~0xFF	High four digits: total number of packages, for example: 1	Example: 0x11
				The lower four bits: the current number of packets, for example: 1	
4	Error type	uint8_t	0x00~0xFF	Parameter error type	
5~6	fat mass	uint16_t	x	Resolution 0.1 , example : 5900 0x0059(89) 8.9	
7~8	Fat percentage	uint16_t	x	Resolution 0.1 , example : 8F00 0x008F(143) 14.3	
9	Fat rate - standard 0	uint8_t	x	Resolution 0.1 , example : 6E 0x6E(110) 11.0	
10~11	Fat percentage - standard 1	uint16_t	x	Resolution 0.1 , example : AA00 0x00AA(170) 17.0	
12~13	Fat percentage - standard 2	uint16_t	x	Resolution 0.1 , example : DC00 0x00DC(220) 22.0	

14~15	Fat percentage - standard 3	uint16_t	x	Resolution 0.1 , example : 0E01 0x010E(270) 27.0
16~17	BMI body mass index	uint16_t	x	Resolution 0.1 , example : D200 0x00D2(210) 21.0
18	BMI body mass index - standard 0	uint8_t	x	Resolution 0.1 , example : B9 0xB9(185) 18.5
19	BMI body mass index - standard 1	uint8_t	x	Resolution 0.1 , example : FA 0xFA(250) 25.0
20~21	BMI body mass index - standard 2	uint16_t	x	Resolution 0.1 , example : 2C01 0x012C(300) 30.0
22~23	basal metabolism	uint16_t	x	Resolution 1 , example : C305 0x05C3(1475) 1475
24~25	Basal Metabolism - Standard 0	uint16_t	x	Resolution 1 , example : 9805 0x0598(1432) 1432
26	physical age	uint8_t	x	Resolution 1 , example : 16 0x16(22) 22
27~28	lean body mass	uint16_t	x	Resolution 0.1 , example : 1602 0x0216(534) 53.4
29~30	Subcutaneous fat mass	uint16_t	x	Resolution 0.1 , example : 5000 0x0050(80) 8.0
31~32	Subcutaneous fat rate	uint16_t	x	Resolution 0.1 , example : 8200 0x0082(130) 13.0
33	Subcutaneous fat rate - standard 0	uint8_t	x	Resolution 0.1 , example : 56 0x56(86) 8.6
34~35	Subcutaneous fat rate - standard 1	uint16_t	x	Resolution 0.1 , example : A700 0x00A7(167) 16.7
36	body score	uint8_t	x	Resolution 1 , example : 60 0x60(96) 96
37	body type	uint8_t	x	Resolution 1 , example : 08 0x08(8) 8
38	Bone mass	uint8_t	x	Resolution 0.1 , example : 1B 0x1B(27) 2.7
39	Bone mass - standard 0	uint8_t	x	Resolution 0.1 , example : 1C 0x1C(28) 2.8
40	Bone mass - standard 1	uint8_t	x	Resolution 0.1 , example : 1E 0x1E(30) 3.0
41~42	ideal weight	uint16_t	x	Resolution 0.1 , example : 8A02 0x028A(650) 65.0
43~44	Moisture rate	uint16_t	x	Resolution 0.1 , example : 4B02 0x024B(587) 58.7
45~46	Moisture content - standard 0	uint16_t	x	Resolution 0.1 , example : 2602 0x0226(550) 55.0
47~48	Moisture rate - standard 1	uint16_t	x	Resolution 0.1 , example : 8A02 0x028A(650) 65.0
49	Visceral fat level	uint8_t	x	Resolution 1 , example : 05 0x05(5) 5
50	Visceral fat level - standard 0	uint8_t	x	Resolution 1 , example : 0A 0x0A(10) 10
51	Visceral fat grade - standard 1	uint8_t	x	Resolution 1 , example : 0F 0x0F(15) 15

52~53	Skeletal muscle mass	uint16_t	x	Resolution 0.1 , example : 1401 0x0114(276) 27.6
54	Skeletal muscle mass - standard 0	uint8_t	x	Resolution 0.1 , example : C8 0xC8(200) 20.0
55~56	Skeletal muscle mass - standard 1	uint16_t	x	Resolution 0.1 , example : 5E01 0x015E(350) 35.0
57~58	protein rate	uint16_t	x	Resolution 0.1 , example : B300 0x00B3(179) 17.9
59	Protein Rate - Standard 0	uint8_t	x	Resolution 0.1 , example : A0 0xA0(160) 16.0
60	Protein Rate - Standard 1	uint8_t	x	Resolution 0.1 , example : B4 0xB4(180) 18.0
61~62	Muscle rate	uint16_t	x	Resolution 0.1 , example : 2D03 0x032D(813) 81.3
63~64	Muscle mass	uint16_t	x	Resolution 0.1 , example : FB01 0x01FB(507) 50.7
65~66	Muscle mass - standard 0	uint16_t	x	Resolution 0.1 , example : EF01 0x01EF(495) 49.5
67~68	Muscle mass - standard 1	uint16_t	x	Resolution 0.1 , example : 5202 0x0252(594) 59.4
69~70	walk	uint16_t	x	Resolution 0.1 , example : 7C00 0x007C(124) 124
71~72	golf	uint16_t	x	Resolution 1 , example : 6D00 0x006D(109) 109
73~74	croquet	uint16_t	x	Resolution 1 , example : 7600 0x0076(118) 118
75~76	Tennis, cycling, basketball	uint16_t	x	Resolution 1 , example : BA00 0x00BA(186) 186
77~78	Squash, bouncy ball, taekwondo, fencing	uint16_t	x	Resolution 1 , example : 3701 0x0137(311) 311
79~80	climb mountains	uint16_t	x	Resolution 1 , example : CB00 0x00CB(203) 203
81~82	Swimming, aerobics, jogging, football, skipping rope	uint16_t	x	Resolution 1 , example : D900 0x00D9(217) 217
83~84	Badminton, table tennis	uint16_t	x	Resolution 1 , example : 8C00 0x008C(140) 140
85	Check Digit	uint8_t	Checksum	Checksum = ~ (frame header + command number +...+ one digit before check digit) +1

TwoLegs:

AA 56 D2 11 00 5D 00 96 00 6E AA 00 DC 00 0E 01 D2 00 B9 FA 2C 01 C3 05 98
05 16 12 02 54 00 88 00 56 A7 00 60 08 1A 1C 1E 8A 02 47 02 26 02 8A 02 05 0A
0F 12 01 C8 5E 01 B2 00 A0 B4 28 03 F8 01 EF 01 52 02 7C 00 6D 00 76 00 BA 00
37 01 CB 00 D9 00 8C 00 81

TwoArms:

AA 56 D2 11 00 5D 00 96 00 6E AA 00 DC 00 0E 01 D2 00 B9 FA 2C 01 C3 05 98
05 16 12 02 54 00 88 00 56 A7 00 60 08 1A 1C 1E 8A 02 47 02 26 02 8A 02 05 0A
0F 12 01 C8 5E 01 B2 00 A0 B4 28 03 F8 01 EF 01 52 02 7C 00 6D 00 76 00 BA 00
37 01 CB 00 D9 00 8C 00 81

7) Parameter read and write 0x10

Parameters can be saved when power is off.

ask

BYTE	type	unit	value	illustrate
0	Frame header	uint8_t	0x55	-
1	Frame length	uint8_t	0x0A	-
2	Order number	uint8_t	0x10	-
3	Read and write	uint8_t	0x01-0x02	0x01 read settings 0x02 write settings
4	Parameter address	uint8_t	x	See parameter description for details
5~8	Parameter value	x	x	Valid only when writing See parameter description for details
9	Check Digit	uint8_t	x	Checksum = ~ (frame header + command number + ... + one digit before check digit) +1

answer

BYTE	type	unit	value	illustrate
0	Frame header	uint8_t	0xAA	-
1	Frame length	uint8_t	0x0A	-
2	Order number	uint8_t	0x10	-
3	Set result	uint8_t	x	Bit0: Read and write error 1 error 0 OK Bit1: Address error 1 error 0 OK Bit2: Data error 1 error 0 OK
4	Parameter address	uint8_t	x	See parameter description for details
5~8	Parameter value	x	x	See parameter description for details
9	Check Digit	uint8_t	x	Checksum = ~ (frame header + command number + ... + one digit before check digit) +1

Parameter Description :

address	Parameter value	default value	Permissions	illustrate
---------	-----------------	---------------	-------------	------------

0x01	0x01:2400 bps 0x02:4800 bps 0x03:9600 bps 0x04:19200 bps 0x05:38400 bps 0x06:57600 bps 0x07:115200 bps 0x08:230400 bps 0x09:460800bps	0x05:38400 bps	R/W	Baud rate : 0x01~0x09 Example: Write baud rate 9600bps Request: 55 0A 10 02 01 03 00 00 00 8B Answer: AA 0A 10 00 01 03 00 00 00 38 Example: Write baud rate 38400bps Request: 55 0A 10 02 01 05 00 00 00 89 Answer: AA 0A 10 00 01 05 00 00 00 36 Example: Read baud rate 38400bps Request: 55 0A 10 01 01 00 00 00 00 8F Answer: AA 0A 10 00 01 05 00 00 00 36
------	---	-------------------	-----	---

8) ISP

Get version information 0xE0/ Enter upgrade 0xE1

ask

BYTE	type	unit	value	illustrate
0	Frame header	uint8_t	0x55	-
1	Frame length	uint8_t	0x05	-
2	Order number	uint8_t	0xE0/0xE1	0xE0 : Get version information 0 xE1 : Enter upgrade
3	app	uint8_t	0x00/0x01/0x02	0x00 : Master 0x01 : Bia 0x02: Weight (Weight is only valid when obtaining version information)
4	Check Digit	uint8_t	x	Checksum = ~ (frame header + command number +...+ one digit before check digit) +1

Get Master version: 5 5 05 E0 00 C6

Get Bia version :5 5 05 E 0 01 C5

Get Weight version: 5 5 05 E 0 02 C4

Enter Master upgrade: 5 5 05 E 1 00 C5

Enter Bia to upgrade :5 5 05 E 1 01 C4

Note: After entering the upgrade, you need to upgrade the program correctly before you can continue to use it. **Send this command with caution** . For detailed upgrade protocol, please refer to "BMH 05108 upgrade flow chart. pdf ".

answer

0xE0 :

BYTE	type	unit	value	illustrate
0	Frame header	uint8_t	0xAA	-
1	Frame length	uint8_t	0x07	-
2	Order number	uint8_t	0xE0	0xE0 : Get version information
3	app	uint8_t	0x00/0x01/0x02	0x00 : Master 0x01 : Bia 0x02: Weight
4~5	version number	uint16_t	-	little endian
6	Check Digit	uint8_t	x	Checksum = ~ (frame header + command number +...+ one digit before check digit) +1

Get Master version: AA 07 E0 00 00 01 6E (V 1.0)

Get Bia version :AA 07 E 1 00 01 01 6C (V 1.1)

Get Weight version: AA 07 E 2 00 02 01 6A (V1.2)

0xE1 :

BYTE	type	unit	value	illustrate
0	Frame header	uint8_t	0xAA	-
1	Frame length	uint8_t	0x07	-
2	Order number	uint8_t	0xE1	0 xE1 : Enter upgrade
3	application	uint8_t	0x00/0x01	0x00 : Master 0x01 : Bia
4	Check Digit	uint8_t	x	Checksum = ~ (frame header + command number +...+ one digit before check digit) +1

Enter Master upgrade: AA 05 E 1 00 70

Enter Bia to upgrade : AA 05 E 1 01 6F

Entering the upgrade is only used during production development.

9) appendix

Weighing status

Senior four	Weighing status	
0	BODY_SCALES_STATE_NULL,	NULL
1	BODY_SCALES_STATE_PREPARE	Preparing for weighing
2	BODY_SCALES_STATE_NOLOAD	No load / current weight < BODY_SCALES_MIN
3	BODY_SCALES_STATE_LOADUP	There is an unlocking weight loaded in the weighing action / locked state, The weight changes from 0 points to > minimum weighing weight
4	BODY_SCALES_STATE_LOAD_OK	Complete a weighing measurement
5	BODY_SCALES_STATE_LOAD_FIX	After completing a weighing measurement, the weight is stable but not unlocked.
6	BODY_SCALES_STATE_LOAD_DOWN	Getting off the scale action
7	BODY_SCALES_STATE_OVERLOAD	Overweight , the current weight is greater than the maximum weighing weight
8	BODY_SCALES_STATE_AUTOON_FAST	Fast ADC automatic weighing judgment
9	BODY_SCALES_STATE_AUTOON_SLOW	Slow ADC automatic scaling judgment
A	BODY_SCALES_STATE_AUTOON_PASS	Automatically uploaded successfully
B	BODY_SCALES_STATE_AUTOON_FAIL	Automatic weighing failed
C	BODY_SCALES_STATE_CAL	Currently in calibration status

Calibration status

Low four bits	Calibration status	
0	BODY_SCALES_CAL_STATUS_CAL0	Calibrating zero point
1	BODY_SCALES_CAL_STATUS_CAL1	Calibrating point 1
2	BODY_SCALES_CAL_STATUS_CAL2	Calibrating point 2
3	BODY_SCALES_CAL_STATUS_CAL3	Calibrating point 3
4	BODY_SCALES_CAL_STATUS_CHECK	Calibration judgment
5	BODY_SCALES_CAL_STATUS_PASS	Calibration successful
6	BODY_SCALES_CAL_STATUS_FAIL	Calibration failed

Impedance measurement status

byte	Impedance measurement status	
0x00	BIA_NULL	Null
0x01	BIA_CHECK_ELECTRODE	Electrode contact detection in progress
0x02	BIA_MEASURE	Under measurement
0x03	BIA_SUCCESS	Measurement successful
0x04	BIA_ERROR_RANGER	Measurement failure - abnormal data, usually caused by abnormal impedance values beyond the range. It may also be caused by abnormal wires and connections (10~1600Ω, recommended single-arm impedance <800Ω)
0x05	BIA_ERROR_REPEAT	Measurement failure - repeated abnormalities , usually posture changes during measurement, etc.
0x06	BIA_USER_EXIT	The measurement was actively exited by the user.

Parameter error type

ERROR_TYPE_NONE	0x00	No errors
ERROR_TYPE_AGE	0x01	Wrong age
ERROR_TYPE_HEIGHT	0x02	Wrong height
ERROR_TYPE_WEIGHT	0x03	Wrong weight
ERROR_TYPE_SEX	0x04	wrong gender
ERROR_TYPE_PEOPLE_TYPE	0x05	User type error
ERROR_TYPE_Z_TWO_LEGS	0x06	Wrong impedance of both feet
ERROR_TYPE_Z_TWO_ARMS	0x07	Hand impedance error
ERROR_TYPE_Z_LEFT_BODY	0x08	Left whole body impedance error
ERROR_TYPE_Z_LEFT_ARM	0x09	Left hand impedance error
ERROR_TYPE_Z_RIGHT_ARM	0x0A	Right hand impedance error
ERROR_TYPE_Z_LEFT_LEG	0x0B	Left foot impedance error
ERROR_TYPE_Z_RIGHT_LEG	0x0C	Right foot impedance error
ERROR_TYPE_Z_TRUNK	0x0D	Torso impedance error

body type

BODY_TYPE_THIN	0x01	Thin type
BODY_TYPE_THIN_MUSCLE	0x02	lean muscular type
BODY_TYPE_MUSCULAR	0x03	muscular type
BODY_TYPE_OBESE_FAT	0x04	bloated obesity type
BODY_TYPE_FAT_MUSCLE	0x05	Fat muscular type
BODY_TYPE_MUSCLE_FAT	0x06	muscular fat
BODY_TYPE_LACK_EXERCISE	0x07	Not athletic
BODY_TYPE_STANDARD	0x08	Standard type
BODY_TYPE_STANDARD_MUSCLE	0x09	Standard muscle type