

# Heatmiser V3 System Protocol

Revision History		
Version	Date	Comments
V3.01	10/11/2009	TFT Profile number added, if the profile number is reset to 0, It will still work with mcv3 and nmv3. Use function number to carry profile number
V3.02	07/04/2010	Added interface for outside temperature and humidity sensor
V3.1	11/11/2011	Add protocol between netmonitor and mc/tft, Add protocol used in Wi-Fi system (single zone)
V3.2	25/11/2011	Add protocol between Multi-link/nm and multi-zone app
V3.3	28/11/2011	Add protocol to read Multi-link logging data
V3.4	29/11/2011	Add protocol to erase history of Wi-Fi stats
V3.5	02/12/2011	Remove the PIN from logging data reply frame, refer to chapter 7.5
V3.6	07/12/2011	1. Change Multi-link/nm dcb structure to add clock, away and boost 2. Add protocol to adjust clock 3. Add protocol to set global away command 4. Add protocol to set hw/timer boost for RS485 prthw/tm1 stat This change made to Chapter 7
V3.7	15/02/2012	1. Add protocol to read all stats summary information from Multi-link/nm to the app, Refer to chapter 7.8
V3.8	20/02/2012	1. Change protocol to support change values to several stats at the same time, read chapter 7.4 for details
V3.9	15/05/2012	Add protocol to Multi-link reply for pin error, refer to 7.3 Add more info to the summary information, refer to 7.8 Reply summary frame for the multi-command frames, refer to 7.4 Add program option value in summary frame, refer to 7.8

1. Introduction. This protocol is used to connect between Heatmiser v3 network products including thermostats (dt-n, dt-en, prt-n, prt-en, prthw-n, tm1-n) and the master controllers the MCV3/TFT, NMV, TR2, Wi-Fi thermostat, Wi-Fi-rf thermostat, and netmonitor sms applications
2. Media. The system is based around half duplex RS485 bus 2 wire data ,  
The frame format is 4800bps, data: 8 bit, start: 1 bit, stop: 1 bit, no parity check.

For backward compatibility TCP interfaces such as the netmonitor and Wi-Fi devices output their data using the RS485 data structure.

Additional commands are available for controlling these intermediate devices.

Data is transferred using a device control block (DCB).

The DCB is a data structure made up of all parameters contained within a thermostat or time clock,

The DCB carries different data and has a different data length for each model of thermostat

Please review DCB tables below for details.

The RS485 half duplex system is a single master multiple slave system and as such when connecting multiple masters care should be taken to avoid device contention.

Note that the MCU used on thermostats with RS485 network is big endian and MCU used on Wi-Fi thermostats is little endian.

### **RS485 network for RS485 thermostat**

#### **3. command Frame format**

##### ***3.1 Frame being sent by master node (MC, NM, TR2)***

Byte order	Comments	Value
0	Destination address	[1,32], 0xff is the broadcast address
1	Frame length, crc included	10(when read) or n+10(when write)
2	Source address	[129,160]
3	Function code	1=write, 0=read
4-5	Start address of dcb to be accessed, low 8 bit	[0, DCB_Len-1], different model of stat has different DCB. DCB means device control block, it is actually all parameters that make up to a data group.
	Start address of dcb to be accessed, high 8 bit	
6-7	Read/write length in byte Low 8 bit	[1, DCB_Len], 0xffff means read the whole dcb
	Read/write length in byte High 8 bit	
8...(n+8)	Contents to be written (n bytes)	If function code is 0, no this segment
N+1	16bitCRC code, low 8 bit	Crc code from 0 to n+8, crc code not included
N+2	16bitCRC code, high 8 bit	

##### ***3.2 Frame reply from slave node (thermostat)***

Byte order	Comments	Value
0	Destination address	[129,160]
1	Frame length, low 8 bit	Crc code included, 7 (when write) or 11+n (when read)
2	Frame length, high 8 bit	
3	Source address	[1,32], broadcast frame has no reply
4	Function code	00. read, 01. write
5-6	Start address of dcb to be accessed, low 8 bit	If function code is 1 (write), no these segments
	Start address of dcb to be accessed, high 8 bit	
7-8	Actual number of read bytes, low 8 bit	
	Actual number of read bytes, high 8 bit	
9...(n+9)	Contents for reading	
N+1	16bit CRC low 8 bit	Crc code of this frame, crc is not included
N+2	16bit CRC high 8 bit	

#### ***Reading and writing a COMMAND frame.***

Only one master can be connected to the RS485 network and originate a session. The RS485 protocol does not allow for bus arbitration, so if more than one master node is connected, data corruption will occur if there is a conflict.

The master node can read all or part of the DCB from a slave node and can also write to some parts of DCB by sending command frames.

#### **Reading**

A Broadcast frame cannot be used for reading data.

We recommend you read the entire DCB for each device providing the master node has enough ram available.

To read a dcb send the frame below (in hex format)

xx 0a \*\* 00 00 00 ff ff ## ##

Where

xx: = destination slave node address, [1, 20]

0a: = frame length

\*\*: = source master node address, [81, a0]

00: = read

00 00: = reading start location, the first byte of the DCB

ff ff: = read the entire data of DCB

## #: low and high 8 bit of 16bit crc.

Please review the crc calculation method at the end of this document.

Any part data of DCB can also be read from a slave node by provide the correct starting address and the number of bytes to read (write in hex format)

To read a part of a DCB send the frame below (in hex format).

xx 0a \*\* 00 s0 s1 n0 n1 ## ##

Where

xx: destination slave node address, [1, 20]

0a: frame length

\*\*: source master node address, [81, a0]

00: read

s0 s1: low and high 8 bit unique address of DCB, the unique address can be seen in DCB tables below.

n0 n1: low and high 8 bit of length in bytes that DCB data will read

## #: low and high 8 bit of 16bit crc checking code.

Care should be taken when reading parts of the DCB, the addresses off all parameters must be contiguous, otherwise there will be no reply.

For example, if we want to read from address 00 to address 32 we must read from 00 to 25 then read 32 with a new command because locations 26 to 31 may not be available on this particular model.

### Writing

Broadcast frames can be sent but do not generate a reply and receipt cannot be guaranteed !

For this reason we recommend that if broadcast frames must be used they should be sent at least 3 times to improve reliability.

Some parameters within the DCB's are read only whereas others are read/write

Sending a write command to a read only address will fail and no reply will be sent.

In the DCB tables below, the read-only parameters are marked in green and read/write parameters are in yellow.

Within the DCB structure parameters are arranged in function groups any data can be read at random but to modify a parameter you must write to the function starting address which is its' lowest address.

For example if we want to modify the frost temperature (address 17) and set the room temperature (address 18), we cannot change them at the same time by sending a single command to modify 2 bytes even though their addresses are contiguous, since they are not in same function group. We must change them one by one.

For those parameters with 2 bytes, we need to send only one command frame with the low 8bit starting address.

For example if we want to modify holiday hours (addresses 24 and 25), we send only one command to write 2 bytes with address of 24 as the starting address.

Comfort levels and time clock settings can only be changed one day at one time by sending a 12 byte frame command.

Here are some sample frames

To Change frost protection to disable send xx 0b \*\* 01 07 00 01 00 00 ## ##

Where

xx: destination slave node address, [1, 20]

0b: frame length

\*\*: source master node address, [81, a0]

01: write

07 00: unique address of frost protection temp in DCB table.

01 00: one byte will be changed

00: frost protection disabled

## ##: low and high 8 bit of 16bit crc checking code.

To Change holiday to 7 days (168 hours, a8 in hex) send. xx 0c \*\* 01 18 00 02 00 a8 00 ## ##

Where

xx: destination slave node address, [1, 20]

0c: frame length

\*\*: source master node address, [81, a0]

01: write

18 00: unique address of holiday in DCB table.

02 00: two bytes will be changed

A8 00: 168 hours for holiday

## ##: low and high 8 bit of 16bit CRC checking code.

To change the comfort levels for Friday to the settings below time

7.00 21

9.00 16

16.00 21

22.00 16

Send

xx 16 \*\* 01 97 00 0c 00 07 00 15 09 00 10 10 00 15 16 00 10 ## #

Where

xx: destination slave node address, [1, 20]

16 frame length

\*\*: source master node address, [81, a0]

01: write

97 00: unique address of Friday comfort level in DCB table.

0c 00: 12 bytes will be changed

07 00 15 09 00 10 10 00 15 16 00 10: comfort level data

## ##: low and high 8 bit of 16bit crc checking code.

## DCB structures for different models of thermostat

### 3.3 DCB for DT/DT-E/PRT/PRT-E

Each model of thermostat uses different DCB structures and so the data returned and the data length will vary.

For example DT/DT-E has 36 bytes,

PRT/PRT-E has 64 bytes in 5/2 day programming mode and has 148 bytes in 7 day programming mode.

For those with 2 bytes parameters, when reading, high 8 bits are sent first while for writing the low 8 bit should be sent first since the MCU used on rs485 model thermostats is in big-endian format

If set temp = 24 in comfort level setting, it means this period is cancelled.

Note that on rs485 network models the data is saved in big-endian format.

Unique address	Index in DCB	comments	Default value(dec)
0	0	DCB length in byte, high 8 bit	
	1	DCB length in byte, low 8 bit	
2	2	Vendor ID 00=HEATMISER, 01=OEM	00
3	3	Bit0~bit6=Version, bit7=floor limit status Bit7=1: currently in floor limit state Bit7=0: currently not in floor limit state	
4	4	Model (DT/DT-E/PRT/PRT-E) (00/01/02/03)	02
5	5	Temperature format, 00= C, 01= F	00
6	6	Switch differential [1,3]	01
7	7	Frost protection disable/enable (R) 0: disable 1: enable	01
8	8	Calibration offset, high 8 bit	00
	9	Calibration offset, low 8 bit	00
10	10	Output delay [0, 15] min	00
11	11	Address [1, 32]	00
12	12	Up down key limit [0, 10]	00
13	13	Sensor selection 00: Built in air sensor only 01: Remote air sensor only 02: Floor sensor only 03: Built in air sensor + floor sensor 04: Remote air sensor + floor sensor	00
14	14	Optimum start [0, 3]	00
15	15	Rate of change	20
16	16	Program mode 00: 5/2 mode 01: 7 day mode	00
17	17	Frost protect temperature [7,17]	12
18	18	Set room temp [5, 35]	16
19	19	Floor max limit [20, 45]	28
20	20	Floor max limit enable/disable, fixed to enable 00: disable 01: enable	01
21	21	On/Off . 00=off, 01=on	00
22	22	Key lock 00=key unlock, 01=key lock	00
23	23	Run mode 00: heating mode (normal mode) 01: frost protection mode	00
24	24	Holiday hours, low 8 bit when write and high 8 bit when read	00
	25	Holiday hours, high 8 bit when write and low 8 bit when read	00
32	26	temp hold minutes, high 8 bit	00
	27	temp hold minutes, low 8 bit high 8 bit ahead when read, low 8 bit ahead when write	00
34	28	Remote air temp, high 8 bit, high 8 bit ahead when read	FF
	29	Remote air temp, low 8 bit. Divided by 10 is the actual temp. Read 0xffff if no sensor connected	FF
36	30	Floor temp, high 8 bit, high 8 bit ahead when read	FF
	31	Floor temp, low 8 bit, Divided by 10 is the actual temp Read 0xFFFF if no sensor connected	FF

38	32	Built in air temp, high 8 bit, high 8 bit ahead when read			FF
	33	Built in air temp, low 8 bit, Divided by 10 is the actual temp. Read 0xFFFF if no sensor connected			FF
40	34	Error code E0: built in air sensor error E1: floor sensor error E2: remote air sensor error			00
41	35	Current heating state 00: no heating currently 01: heating currently <b>DCB finish here for DT/DT-E</b>			00
43	36	Week 1~7=Mon. ~ Sun.			FF
	37	Hour [0,23]			FF
	38	Min [0,59]			FF
	39	Sec [0,59]			FF
47	40	weekday	Time1	Hour [0, 24]	07
				Min 00, 30	00
				Set temp	21
			Time2	Hour	09
				Min	00
				Set temp	16
			Time3	Hour	16
				Min	00
				Set temp	21
			Time4	Hour	22
				Min	00
				Set temp	16
59	52	weekend	Time1	Hour	09
				Min	00
				Set temp	21
			Time2	Hour	22
				Min	00
				Set temp	16
			Time3	Hour	24
				Min	00
				Set temp	16
			Time4	Hour	24
				Min	00
	63			Set temp <b>Dcb finish here if program mode is 00 (5/2 day)</b>	16
103	64	Mon	Time1	Hour	07
				Min	00
				Set temp	21
			Time2	Hour	09
				Min	00
				Set temp	16
			Time3	Hour	16
				Min	00
				Set temp	21
			Time4	Hour	22
				Min	00
				Set temp	16
115	76	Tue	Time1	Hour	07
				Min	00

			Time2	Set temp	21
				Hour	09
				Min	00
			Time3	Set temp	16
				Hour	16
				Min	00
			Time4	Set temp	21
				Hour	22
				Min	00
				Set temp	16
127	88			Hour	07
				Min	00
		Wed	Time1	Set temp	21
				Hour	09
				Min	00
			Time2	Set temp	16
				Hour	16
				Min	00
			Time3	Set temp	21
				Hour	22
				Min	00
			Time4	Set temp	16
				Hour	22
				Min	00
		Thu	Time1	Set temp	21
139	100			Hour	07
				Min	00
			Time2	Set temp	16
				Hour	09
				Min	00
			Time3	Set temp	21
				Hour	16
				Min	00
			Time4	Set temp	16
				Hour	22
				Min	00
		Fri	Time1	Set temp	21
151	112			Hour	07
				Min	00
			Time2	Set temp	16
				Hour	09
				Min	00
			Time3	Set temp	21
				Hour	16
				Min	00
			Time4	Set temp	16
				Hour	22
				Min	00
		Sat	Time1	Set temp	21
163	124			Hour	09
				Min	00
			Time2	Set temp	16
				Hour	22
				Min	00
			Time3	Set temp	16
				Hour	24
				Min	00

			Time4	Hour	24
				Min	00
				Set temp	16
175	136		Time1	Hour	09
				Min	00
				Set temp	21
			Time2	Hour	22
				Min	00
				Set temp	16
		Sun	Time3	Hour	24
				Min	00
				Set temp	16
			Time4	Hour	24
				Min	00
	147			Set temp	16

### 3.4 DCB for PRTHW

If hour=24 in programming of comfort level and timer, it means this period is cancelled

Unique address	Index in dcb	Comments	Default value (dec)
0	0	DCB length in byte, high 8 bit	
	1	DCB length in byte, low 8 bit	
2	2	Vendor ID 00=HEATMISER, 01=OEM	00
3	3	Version	10
4	4	Model (PRTHW=04)	04
5	5	Temperature format, 00= C, 01= F	00
6	6	Switch differential [1,3]	01
7	7	Frost protection disable/enable (R) 0: disable 1: enable	01
8	8	Calibration offset, high 8 bit	00
	9	Calibration offset, low 8 bit	00
10	10	Output delay [0, 15] min	00
11	11	Address [1, 32]	00
12	12	Up down key limit [0, 10]	00
13	13	Sensor selection 00: Built in air sensor only	00
14	14	Optimum start [0, 3]	00
15	15	Rate of change	20
16	16	Program mode 00: 5/2 mode 01: 7 day mode	00
17	17	Frost protect temperature [7,17]	12
18	18	Set room temp [5, 35]	16
19	19	Floor max limit [20, 45]	28
20	20	Floor max limit enable/disable (no use) 00: disable 01: enable	01
21	21	On/Off . 00=off, 01=on	00
22	22	Key lock 00=key unlock, 01=key lock	00
23	23	Run mode 00: heating mode (normal mode) 01: frost protection mode	00
24	24	Holiday hours, low 8 bit when write and high 8 bit when read	00



	25	Holiday hours, high 8 bit when write and low 8 bit when read			00
32	26	temp hold minutes, high 8 bit			00
	27	temp hold minutes, low 8 bit high 8 bit ahead when read, low 8 bit ahead when write			00
34	28	Remote air temp, high 8 bit, high 8 bit ahead when read			FF
	29	Remote air temp, low 8 bit. Divided by 10 is the actual temp. Read 0xffff if no sensor connected			FF
36	30	Floor temp, high 8 bit, high 8 bit ahead when read			FF
	31	Floor temp, low 8 bit, Divided by 10 is the actual temp Read 0xFFFF if no sensor connected			FF
38	32	Built in air temp, high 8 bit, high 8 bit ahead when read			FF
	33	Built in air temp, low 8 bit, Divided by 10 is the actual temp. Read 0xFFFF if no sensor connected			FF
40	34	Error code E0: built in air sensor error E1: floor sensor error E2: remote air sensor error			00
41	35	Current heating state 00: no heating currently 01: heating currently			00
42	36	Current Hot water state Read 00: hot water is off now Read 01: hot water is on now Write 00, let hw run based on programmed timer period Write 01, override hot water to on Write 02, override hot water to off			00
43	37	Week 1~7=Mon. ~ Sun			FF
	38	Hour [0, 23]			FF
	39	Min [0,59]			FF
	40	Sec [0, 59]			FF
47	41	weekday	Time1	Hour	07
				Min	00
				Set temp	21
			Time2	Hour	09
				Min	00
				Set temp	16
			Time3	Hour	16
				Min	00
				Set temp	21
			Time4	Hour	22
				Min	00
				Set temp	16
59	53	weekend	Time1	Hour	09
				Min	00
				Set temp	21
			Time2	Hour	22
				Min	00
				Set temp	16
			Time3	Hour	24
				Min	00
				Set temp	16
			Time4	Hour	24
				Min	00
				Set temp	16
71	65	Weekday	Time1 / On	Hour	07
				Min	00

			Time1 / Off	Hour	09
				Min	00
			Time2 / On	Hour	16
				Min	00
			Time2 / Off	Hour	20
				Min	00
			Time3 / On	Hour	24
				Min	00

Time3 / Off

				Set temp	16
127	121	Wed	Time1	Hour	07
				Min	00
				Set temp	21
			Time2	Hour	09
				Min	00
				Set temp	16
			Time3	Hour	16
				Min	00
				Set temp	21
			Time4	Hour	22
				Min	00
				Set temp	16
139	133	Thu	Time1	Hour	07
				Min	00
				Set temp	21
			Time2	Hour	09
				Min	00
				Set temp	16
			Time3	Hour	16
				Min	00
				Set temp	21
			Time4	Hour	22
				Min	00
				Set temp	16
151	145	Fri	Time1	Hour	07
				Min	00
				Set temp	21
					09
			Time2	Min	00
				Set temp	16
			Time3		

			Time3	Hour	24
				Min	00
				Set temp	16
			Time4	Hour	24
				Min	00
				Set temp	16
187	181	Monday	Time1 / On	Hour	07
				Min	00
			Time1 / Off	Hour	09
				Min	00
			Time2 / On	Hour	16
				Min	00
			Time2 / Off	Hour	20
				Min	00
			Time3 / On	Hour	24
				Min	00
			Time3 / Off	Hour	24
				Min	00
			Time4 / On	Hour	24
				Min	00
			Time4 / Off	Hour	24
				Min	00
203	197	Tue	Time1 / On	Hour	07
				Min	00
			Time1 / Off	Hour	09
				Min	00
			Time2 / On	Hour	16
				Min	00
			Time2 / Off	Hour	20
				Min	00
			Time3 / On	Hour	24
				Min	00
			Time3 / Off	Hour	24
				Min	00
			Time4 / On	Hour	24
				Min	00
			Time4 / Off	Hour	24
				Min	00
219	213	Wed	Time1 / On	Hour	07
				Min	00
			Time1 / Off	Hour	09
				Min	00
			Time2 / On	Hour	16
				Min	00
			Time2 / Off	Hour	20
				Min	00
			Time3 / On	Hour	24
				Min	00
			Time3 / Off	Hour	24
				Min	00
			Time4 / On	Hour	24
				Min	00
			Time4 / Off	Hour	24
				Min	00
235	229	Thu	Time1 / On	Hour	07

				Min	00
			Time1 / Off	Hour	09
				Min	00
			Time2 / On	Hour	16
				Min	00
			Time2 / Off	Hour	20
				Min	00
			Time3 / On	Hour	24
				Min	00
			Time3 / Off	Hour	24
				Min	00
			Time4 / On	Hour	24
				Min	00
			Time4 / Off	Hour	24
				Min	00
251	245	Fri	Time1 / On	Hour	07
				Min	00
			Time1 / Off	Hour	09
				Min	00
			Time2 / On	Hour	16
				Min	00
			Time2 / Off	Hour	20
				Min	00
			Time3 / On	Hour	24
				Min	00
			Time3 / Off	Hour	24
				Min	00
			Time4 / On	Hour	24
				Min	00
			Time4 / Off	Hour	24
				Min	00
267	261	Sat	Time1 / On	Hour	07
				Min	00
			Time1 / Off	Hour	09
				Min	00
			Time2 / On	Hour	16
				Min	00
			Time2 / Off	Hour	20
				Min	00
			Time3 / On	Hour	24
				Min	00
			Time3 / Off	Hour	24
				Min	00
			Time4 / On	Hour	24
				Min	00
			Time4 / Off	Hour	24
				Min	00
283	277	Sun	Time1 / On	Hour	07
				Min	00
			Time1 / Off	Hour	09
				Min	00
			Time2 / On	Hour	16
				Min	00
			Time2 / Off	Hour	20
				Min	00

			Time3 / On	Hour	24
				Min	00
			Time3 / Off	Hour	24
				Min	00
			Time4 / On	Hour	24
				Min	00
			Time4 / Off	Hour	24
	292			Min	00

### 3.5 DCB for TM1

Unique address	Index in DCB	Comments			Default value (dec)
0	0	DCB length in byte, high 8 bit			
	1	DCB length in byte, low 8 bit			
2	2	Vendor ID 00: Heatmiser 01: OEM			00
3	3	Version 10 = VER 1.0			10
4	4	Model (TM1 = 5)			05
11	5	Address [1, 32]			00
16	6	Program mode 00: 5/2 mode 01: 7 day mode 02: Countdown timer mode			00
	7	Reserved (not used currently)			
21	8	On/Off (R/W) 00: off, 01: on			00
22	9	Key lock (R/W) 00: key unlock, 01: key lock			00
24	10	Holiday hours, high 8 bit (R/W)			00
	11	Holiday hours, low 8 bit (R/W) High byte first when read, low byte first when write			00
26	12	Countdown min, high 8 bit			[0,1800] min
	13	Countdown min, low 8 bit High 8 bit ahead when read Low 8 bit ahead when write			
42	14	Current timer state(R/W) Read 00: timer is off now Read 01: timer is on now Write 00, let timer run based on programmed timer period Write 01, override timer to on Write 02, override timer to off			00
43	15	Week 1~7 = Mon. ~ Sun			
	16	Hour [0, 23]			
	17	Min [0, 59]			
	18	Sec [0, 59] <b>DCB finish here if program mode is 02 (countdown timer mode)</b>			
71	19	Weekday	Time1 / On	Hour	07
				Min	00
			Time1 / Off	Hour	09
				Min	00
			Time2 / On	Hour	16
				Min	00
			Time2 / Off	Hour	20
				Min	00

			Time3 / On	Hour	24
				Min	00
			Time3 / Off	Hour	24
				Min	00
			Time4 / On	Hour	24
				Min	00
			Time4 / Off	Hour	24
				Min	00
87	35	Weekend	Time1 / On	Hour	07
				Min	00
			Time1 / Off	Hour	09
				Min	00
			Time2 / On	Hour	16
				Min	00
			Time2 / Off	Hour	20
				Min	00
			Time3 / On	Hour	24
				Min	00
			Time3 / Off	Hour	24
				Min	00
			Time4 / On	Hour	24
				Min	00
			Time4 / Off	Hour	24
	50			Min <b>Dec finish here if program mode is 00 (5/2 day)</b>	00
187	51	Monday	Time1 / On	Hour	07
				Min	00
			Time1 / Off	Hour	09
				Min	00
			Time2 / On	Hour	16
				Min	00
			Time2 / Off	Hour	20
				Min	00
			Time3 / On	Hour	24
				Min	00
			Time3 / Off	Hour	24
				Min	00
			Time4 / On	Hour	24
				Min	00
			Time4 / Off	Hour	24
				Min	00
203	67	Tue	Time1 / On	Hour	07
				Min	00
			Time1 / Off	Hour	09
				Min	00
			Time2 / On	Hour	16
				Min	00
			Time2 / Off	Hour	20
				Min	00
			Time3 / On	Hour	24
				Min	00
			Time3 / Off	Hour	24
				Min	00
			Time4 / On	Hour	24

				Min	00
			Time4 / Off	Hour	24
				Min	00
219	83	Wed	Time1 / On	Hour	07
				Min	00
			Time1 / Off	Hour	09
				Min	00
			Time2 / On	Hour	16
				Min	00
			Time2 / Off	Hour	20
				Min	00
			Time3 / On	Hour	24
				Min	00
			Time3 / Off	Hour	24
				Min	00
			Time4 / On	Hour	24
				Min	00
			Time4 / Off	Hour	24
				Min	00
235	99	Thu	Time1 / On	Hour	07
				Min	00
			Time1 / Off	Hour	09
				Min	00
			Time2 / On	Hour	16
				Min	00
			Time2 / Off	Hour	20
				Min	00
			Time3 / On	Hour	24
				Min	00
			Time3 / Off	Hour	24
				Min	00
			Time4 / On	Hour	24
				Min	00
			Time4 / Off	Hour	24
				Min	00
251	115	Fri	Time1 / On	Hour	07
				Min	00
			Time1 / Off	Hour	09
				Min	00
			Time2 / On	Hour	16
				Min	00
			Time2 / Off	Hour	20
				Min	00
			Time3 / On	Hour	24
				Min	00
			Time3 / Off	Hour	24
				Min	00
			Time4 / On	Hour	24
				Min	00
			Time4 / Off	Hour	24
				Min	00
267	131	Sat	Time1 / On	Hour	07
				Min	00
			Time1 / Off	Hour	09
				Min	00



			Time2 / On	Hour	16
				Min	00
			Time2 / Off	Hour	20
				Min	00
			Time3 / On	Hour	24
				Min	00
			Time3 / Off	Hour	24
				Min	00
			Time4 / On	Hour	24
				Min	00
			Time4 / Off	Hour	24
				Min	00
283	147	Sun	Time1 / On	Hour	07
				Min	00
			Time1 / Off	Hour	09
				Min	00
			Time2 / On	Hour	16
				Min	00
			Time2 / Off	Hour	20
				Min	00
			Time3 / On	Hour	24
				Min	00
			Time3 / Off	Hour	24
				Min	00
			Time4 / On	Hour	24
				Min	00
			Time4 / Off	Hour	24
	162			Min	00

### 3.6 DCB for HC-EN V3

If hour=24 in programming of comfort level and timer, it means this period is cancelled, should read the entire dcb at once

Network address	Index in dcb	Comments	Default value (dec)
0	0	DCB length in byte, high 8 bit	
	1	DCB length in byte, low 8 bit	
2	2	Vendor ID 00=HEATMISER, 01=OEM	00
3	3	Version	10
4	4	Model (HC-EN=07)	04
5	5	Temperature format, 00= C, 01= F	00
11	6	Address [1, 32]	01
	7	Humidity sensor address	01
12	8	Up down key limit [0, 10]	00
13	9	Sensor selection 00: Built in air sensor only 01: Remote air sensor only 02: Floor sensor only 03: Built in air sensor + floor sensor 04: Remote air sensor + floor sensor	00
14	10	Optimum start [0, 3]	00
7	11	Frost protection disable/enable (R) 0: disable 1: enable	00

16	12	Program mode 00: 5/2 mode 01: 7 day mode		00	
29	13	Cooling Enable 00: Disable, 01=Enable		01	
	14	Dew point limit differential, [0,10]		03	
	15	Dew point time limit in minute, [0,60]		0	
8	16	Calibration offset, high 8 bit		00	
	17	Calibration offset, low 8 bit		12	
24	18	Holiday hours, low 8 bit when write and high 8 bit when read		16	
	19	Holiday hours, high 8 bit when write and low 8 bit when read		28	
10	20	Output delay [0, 15] min		01	
6	21	Switch differential [1,3]		00	
	22	Floor max limit [20, 45]		00	
	23	Floor min limit [10, 30]		00	
27	24	Cooling set temp		00	
17	25	Frost protect temperature [7,17]		00	
15	26	Rate of change		00	
21	27	On/Off . 00=off, 01=on		00	
23	28	Run mode 00: heating mode 01: frost protection mode 02: cooling mode		00	
22	29	Key lock 00=key unlock, 01=key lock		FF	
32	30	temp hold minutes, high 8 bit		FF	
	31	temp hold minutes, low 8 bit high 8 bit ahead when read, low 8 bit ahead when write		FF	
38	32	Built-in/remote air temp, high 8 bit, high 8 bit ahead when read		FF	
	33	Built-in/remote air temp, low 8 bit, Divided by 10 is the actual temp. Read 0xFFFF if no sensor connected		FF	
36	34	Floor temp, high 8 bit, high 8 bit ahead when readE2: remote air sensor error		00	
	35	Floor temp, low 8 bit, Divided by 10 is the actual temp Read 0xFFFF if no sensor connected00: no heating currently 01: heating currently		00	
30	36	Humidity RH%		00	
28	37	Dew Point temp			
41	38	Output state bit0=1=heat on, bit1=1=heat floor limit bit4=1=cool on, bit5=1=cool floor limit reach bit6=1=dew point reach bit7=1=during dew point exceeded limit period			
18	39	Heating set temp			
40	40	Error code E0: built in air sensor error E1: floor sensor error E2: remote air sensor error			
43	41	Week 1~7=Mon. ~ Sun		FF	
	42	Hour [0, 23]		FF	
	43	Min [0,59]		FF	
	44	Sec [0, 59]		FF	
47	45	weekday	Hour	07	
			Time1	Min	00
				Set temp	21
			Time2	Hour	09

				Min	00
				Set temp	16
			Time3	Hour	16
				Min	00
				Set temp	21
			Time4	Hour	22
				Min	00
				Set temp	16
59	57		weekend	Hour	09
				Min	00
				Set temp	21
				Hour	22
				Min	00
				Set temp	16
				Hour	24
				Min	00
				Set temp	16
				Hour	24
				Min	00
				Set temp	16
103	69	Mon	Time1	Hour	07
				Min	00
				Set temp	21
			Time2	Hour	09
				Min	00
				Set temp	16
			Time3	Hour	16
				Min	00
				Set temp	21
			Time4	Hour	22
				Min	00
				Set temp	16
115	81	Tue	Time1	Hour	07
				Min	00
				Set temp	21
			Time2	Hour	09
				Min	00
				Set temp	16
			Time3	Hour	16
				Min	00
				Set temp	21
			Time4	Hour	22
				Min	00
				Set temp	16
127	93	Wed	Time1	Hour	07
				Min	00
				Set temp	21
			Time2	Hour	09
				Min	00
				Set temp	16
			Time3	Hour	16
				Min	00
				Set temp	21
			Time4	Hour	22
				Min	00

				Set temp	16
139	105	Thu	Time1	Hour	07
				Min	00
				Set temp	21
			Time2	Hour	09
				Min	00
				Set temp	16
			Time3	Hour	16
				Min	00
				Set temp	21
			Time4	Hour	22
				Min	00
				Set temp	16
151	117	Fri	Time1	Hour	07
				Min	00
				Set temp	21
			Time2	Hour	09
				Min	00
				Set temp	16
			Time3	Hour	16
				Min	00
				Set temp	21
			Time4	Hour	22
				Min	00
				Set temp	16
163	129	Sat	Time1	Hour	09
				Min	00
				Set temp	21
			Time2	Hour	22
				Min	00
				Set temp	16
			Time3	Hour	24
				Min	00
				Set temp	16
			Time4	Hour	24
				Min	00
				Set temp	16
175	141	Sun	Time1	Hour	09
				Min	00
				Set temp	21
			Time2	Hour	22
				Min	00
				Set temp	16
			Time3	Hour	24
				Min	00
				Set temp	16
			Time4	Hour	24
				Min	00
				Set temp	16
	152			Min	00

### 3.7 DCB for Outside Temperature Sensor (OTS), read the entire dcb at once

Write address	Index in DCB	comments	Default value(dec)
	0	DCB length in byte, high 8 bit	00

	1	DCB length in byte, low 8 bit	
	2	Vendor ID	00
	3	Version	10
	4	Model (OTS=20)	21
	5	Address=81	81
	6,7	Temperature	00
8	8,9	Calibration offset	00
	10	Error code, 0x00=ok, 0xe0=error	00=no error 0xe0=error

### 3.8 DCB for Humidity Temperature Sensor (HTS), read the entire dcb at once

Write address	Index in DCB	comments	Default value(dec)
	0	DCB length in byte, high 8 bit	00
	1	DCB length in byte, low 8 bit	22
	2	Vendor ID	00
	3	Version	10
	4	Model (HTS=22)	22
	5	Address [41,72]	0
6	6,7	Probe calibration offset	00
8	8,9	Built-in temp sensor calibration offset	00
10	10,11	Humidity sensor calibration offset	00
	12,13	Probe temperature	00
	14,15	Built-in sensor temperature	00
	16,17	Humidity value	00
	18,19	Calculated dew point	00
	20	Built-in sensor err code, 0xe0=temp, 0xe1=humidity, 0x00=ok	0
	15	Probe err code, 0xe0=error, 0x00=ok	0

## 4. Protocol between NM and MC/TFT

4.1 The frame format that the NM uses when reading from the MC/TFT is the same as that used by MC/TFT when reading from the stat command frame, but the destination address is 0x81 which is the MC/TFT comms number, the reply contents are

index	content	range	
0	Holiday return year	[0,99]	
1	Holiday return month	[1,12]	
2	Holiday return day of month	[1,31]	
3	Holiday return hour	[0,23]	
4	Holiday return min	[0,59]	
5	Holiday return sec	0	
6	Away info	1=away from mc 2=away from nm 3=holiday from mc 4=holiday from nm 5=home from mc 6=home from nm	
7	Comms number	[1,32]	Mostly 32 items, 0 item is possible
8	Boost hour	[0,30]	
...			
	Comms number	[1,32]	
	Boost hour	[0,30]	

4.2 The frame format of the NM when writing to the MC is the same as the one used when the MC is writing to the stat but uses 0x81 as the destination address, the writing contents are

Write address	Content	Values
100	Clock	7 bytes, in order they are Year, month, day, week, hour, min, sec
101	Boost or cancel boost from mc	2 bytes, in order they are comms number and boost hours, zero boost hour means cancel boost
70	Away info	1 byte 2=away from nm 4=holiday from nm 5=home from mc

### 5. The Wi-Fi system protocol (single zone)

The Wi-Fi thermostat protocol is a little different than the RS485 network protocol,

The Wi-Fi thermostat and its iphone application (one zone system) communicate over tcp with port **8068**,

**NOTE the dcb of Wi-Fi thermostat is in little-endian format**

#### 5.1 Reading frame of command (read the entire dcb at once)

Byte No	Contents	Value	Comments
0	Frame head	0x93	
1	Frame len, low 8 bit		
2	Frame len, high 8 bit		
3	PIN, low 8 bit		
4	PIN, hi 8 bit		
5	Start address of dcb low 8bit		If start address=0x0000 and number of bytes=0xffff means read the entire dcb
6	Start address of dcb high 8bit		
7	Number of bytes to be read low 8bit		
8	Number of bytes to be read high 8bit		
9	16bit crc low 8bit		
10	16bit crc high 8bit		

#### 5.2 Reply from Wi-Fi thermostat

Byte No	Contents	Value	Comments
0	Frame head	0x94	
1	Frame len, low 8 bit		
2	Frame len, high 8 bit		
3	Start address low 8bit		
4	Start address high 8bit		
5	Number of bytes to be read low 8bit	0x0000 means pin error	
6	Number of bytes to be read high 8bit		
...	Contents of dcb		
	16bit crc low 8bit		
	16bit crc high 8bit		

#### 5.3 When using the iphone app to write to the stat, use the following improved protocol to reduce I phone power consumption

Byte No.	Contents	Value	Comments
0	Frame head	0xa3	
1	Frame len, low 8 bit		
2	Frame len, high 8 bit		
3	Pin, low 8 bit		
4	Pin, high 8 bit		
5	Total number of writing items		
6	1 <sup>st</sup> item address low 8 bit		
7	1 <sup>st</sup> item address high 8 bit		

8	1 <sup>st</sup> item bytes number to be written		
9	1 <sup>st</sup> item contents to be written		
10	2 <sup>st</sup> item address low 8 bit		
11	2 <sup>st</sup> item address high 8 bit		
12	2 <sup>st</sup> item bytes number		
13	2 <sup>st</sup> item contents to be written		
...	...		
	16bit crc low 8 bit		
	16bit crc high 8 bit		

5.4 Reply from the Wi-Fi thermostat is the same as the reply for the inquiry command

## 6. DCB of the Wi-Fi thermostats

DCB of the Wi-Fi thermostats are a little different from that of RS485 ones

6.1 DCB for Wi-Fi DT/DT-E/PRT/PRT-E WI-FI / WI-FI-RF, read the entire dcb at once

Unique address	Index in DCB	comments	Default value(dec)
0	0	DCB length in byte, high 8 bit	
	1	DCB length in byte, low 8 bit	
2	2	Vendor ID 00=HEATMISER, 01=OEM	00
3	3	Bit0~bit6=Version, bit7=floor limit status Bit7=1: currently in floor limit state Bit7=0: currently not in floor limit state	
4	4	Model (DT/DT-E/PRT/PRT-E) (00/01/02/03)	02
5	5	Temperature format, 00= C, 01= F	00
6	6	Switch differential [1,3]	01
7	7	Frost protection disable/enable (R) 0: disable 1: enable	01
8	8	Calibration offset, high 8 bit	00
	9	Calibration offset, low 8 bit	00
10	10	Output delay [0, 15] min	00
11	11	Address [1, 32], not used	00
12	12	Up down key limit [0, 10]	00
13	13	Sensor selection 00: Built in air sensor only 01: Remote air sensor only 02: Floor sensor only 03: Built in air sensor + floor sensor 04: Remote air sensor + floor sensor	00
14	14	Optimum start [0, 3]	00
15	15	Rate of change	20
16	16	Program mode 00: 5/2 mode 01: 7 day mode	00
17	17	Frost protect temperature [7,17]	12
18	18	Set room temp [5, 35]	16
19	19	Floor max limit [20, 45]	28
20	20	Floor max limit enable/disable, fixed to enable 00: disable 01: enable	01
21	21	On/Off . 00=off, 01=on	00
22	22	Key lock 00=key unlock, 01=key lock	00
23	23	Run mode 00: heating mode (normal mode) 01: frost protection mode	00

31	24	Away mode (not used here)			
24	25	Holiday return date, year			Write holiday return date&time to issue a holiday, write one byte 00 to cancel it
	26	Holiday return date, month			
	27	Holiday return date, day of month			
	28	Holiday return time, hour			
	29	Holiday return time, minute			
24	30	Holiday Enabled 00 = not in holiday mode currently for reading, if write 00 will cancel holiday 01 = currently in holiday mode, writing will not accept 01 value			
32	31	temp hold minutes, high 8 bit			00
	32	temp hold minutes, low 8 bit high 8 bit ahead when read, low 8 bit ahead when write			00
34	33	Remote air temp, high 8 bit, high 8 bit ahead when read			FF
	34	Remote air temp, low 8 bit. Divided by 10 is the actual temp. Read 0xffff if no sensor connected			FF
36	35	Floor temp, high 8 bit, high 8 bit ahead when read			FF
	36	Floor temp, low 8 bit, Divided by 10 is the actual temp Read 0xFFFF if no sensor connected			FF
38	37	Built in air temp, high 8 bit, high 8 bit ahead when read			FF
	38	Built in air temp, low 8 bit, Divided by 10 is the actual temp. Read 0xFFFF if no sensor connected			FF
40	39	Error code E0: built in air sensor error E1: floor sensor error E2: remote air sensor error			00
41	40	Current heating state 00: no heating currently 01: heating currently <b>DCB finish here for DT/DT-E</b>			00
43	41	Year [00,99]			
	42	Month [1, 12]			
	43	Day of month [1, 31]			
	44	Week 1~7=Mon. ~ Sun.			FF
	45	Hour [0,23]			FF
	46	Min [0,59]			FF
	47	Sec [0,59]			FF
47	48	weekday	Time1	Hour [0, 24]	07
				Min 00, 30	00
				Set temp	21
			Time2	Hour	09
				Min	00
				Set temp	16
			Time3	Hour	16
				Min	00
				Set temp	21
			Time4	Hour	22
				Min	00
				Set temp	16
59	60	weekend	Time1	Hour	09
				Min	00
				Set temp	21
			Time2	Hour	22
				Min	00
				Set temp	16



			Time3	Hour	24
				Min	00
				Set temp	16
			Time4	Hour	24
				Min	00
	71			Set temp <b>Dcb finish here if program mode is 00 (5/2 day)</b>	16
103	72	Mon	Time1	Hour	07
				Min	00
				Set temp	21
			Time2	Hour	09
				Min	00
				Set temp	16
			Time3	Hour	16
				Min	00
				Set temp	21
			Time4	Hour	22
				Min	00
				Set temp	16
115	84	Tue	Time1	Hour	07
				Min	00
				Set temp	21
			Time2	Hour	09
				Min	00
				Set temp	16
			Time3	Hour	16
				Min	00
				Set temp	21
			Time4	Hour	22
				Min	00
				Set temp	16
127	96	Wed	Time1	Hour	07
				Min	00
				Set temp	21
			Time2	Hour	09
				Min	00
				Set temp	16
			Time3	Hour	16
				Min	00
				Set temp	21
			Time4	Hour	22
				Min	00
				Set temp	16
139	108	Thu	Time1	Hour	07
				Min	00
				Set temp	21
			Time2	Hour	09
				Min	00
				Set temp	16
			Time3	Hour	16
				Min	00
				Set temp	21
			Time4	Hour	22
				Min	00
				Set temp	00

				Set temp	16
151	120	Fri	Time1	Hour	07
				Min	00
				Set temp	21
			Time2	Hour	09
				Min	00
				Set temp	16
			Time3	Hour	16
				Min	00
				Set temp	21
			Time4	Hour	22
				Min	00
				Set temp	16
163	132	Sat	Time1	Hour	09
				Min	00
				Set temp	21
			Time2	Hour	22
				Min	00
				Set temp	16
			Time3	Hour	24
				Min	00
				Set temp	16
			Time4	Hour	24
				Min	00
				Set temp	16
175	144	Sun	Time1	Hour	09
				Min	00
				Set temp	21
			Time2	Hour	22
				Min	00
				Set temp	16
			Time3	Hour	24
				Min	00
				Set temp	16
			Time4	Hour	24
				Min	00
				Set temp	16
	155			Set temp	16

6.2 DCB for PRTHW WI-FI / WI-FI-RF, read the entire dcb at once

If hour=24 in programming of comfort level and timer, it means this period is cancelled

Unique address	Index in dcb	Comments	Default value (dec)
0	0	DCB length in byte, high 8 bit	
	1	DCB length in byte, low 8 bit	
2	2	Vendor ID 00=HEATMISER, 01=OEM	00
3	3	Version	10
4	4	Model (PRTHW=04)	04
5	5	Temperature format, 00= C, 01= F	00
6	6	Switch differential [1,3]	01
7	7	Frost protection disable/enable (R) 0: disable 1: enable	01
8	8	Calibration offset, high 8 bit	00
	9	Calibration offset, low 8 bit	00

10	10	Output delay [0, 15] min	00
11	11	Address [1, 32], not used	00
12	12	Up down key limit [0, 10]	00
13	13	Sensor selection 00: Built in air sensor only	00
14	14	Optimum start [0, 3]	00
15	15	Rate of change	20
16	16	Program mode 00: 5/2 mode 01: 7 day mode	00
17	17	Frost protect temperature [7,17]	12
18	18	Set room temp [5, 35]	16
19	19	Floor max limit [20, 45], not used	28
20	20	Floor max limit enable/disable (no use) 00: disable 01: enable	01
21	21	On/Off . 00=off, 01=on	00
22	22	Key lock 00=key unlock, 01=key lock	00
23	23	Run mode 00: heating mode (normal mode) 01: frost protection mode	00
31	24	Away mode 00 = not away mode, 01 = away mode	
24	25	Holiday return date, year	Write holiday return date&time to issue a holiday, write one byte 00 to cancel it
	26	Holiday return date, month	
	27	Holiday return date, day of month	
	28	Holiday return time, hour	
	29	Holiday return time, minute	
24	30	Holiday Enabled 00 = not in holiday mode currently for reading, if write 00 will cancel holiday 01 = currently in holiday mode, writing will not accept 01 value	
32	31	temp hold minutes, high 8 bit	00
	32	temp hold minutes, low 8 bit high 8 bit ahead when read, low 8 bit ahead when write	00
34	33	Remote air temp, high 8 bit, high 8 bit ahead when read	FF
	34	Remote air temp, low 8 bit. Divided by 10 is the actual temp. Read 0xffff if no sensor connected	FF
36	35	Floor temp, high 8 bit, high 8 bit ahead when read	FF
	36	Floor temp, low 8 bit, Divided by 10 is the actual temp Read 0xFFFF if no sensor connected	FF
38	37	Built in air temp, high 8 bit, high 8 bit ahead when read	FF
	38	Built in air temp, low 8 bit, Divided by 10 is the actual temp. Read 0xFFFF if no sensor connected	FF
40	39	Error code E0: built in air sensor error E1: floor sensor error E2: remote air sensor error	00
41	40	Current heating state 00: no heating currently 01: heating currently	00
25	41	Boost in min high byte (writing function works in 1.6 code)	
	42	Boost in min low byte	
42	43	Current Hot water state Read 00: hot water is off now	00

		Read 01: hot water is on now Write 00, let hw run based on programmed timer period Write 01, override hot water to on Write 02, override hot water to off			
43	44	Year, [00,99]			
	45	Month, [1,12]			
	46	Day of month, [1,31]			
	47	Week 1~7=Mon. ~ Sun			FF
	48	Hour [0, 23]			FF
	49	Min [0,59]			FF
	50	Sec [0, 59]			FF
47	51	weekday	Time1	Hour	07
				Min	00
				Set temp	21
			Time2	Hour	09
				Min	00
				Set temp	16
			Time3	Hour	16
				Min	00
				Set temp	21
			Time4	Hour	22
				Min	00
				Set temp	16
59	63	weekend	Time1	Hour	09
				Min	00
				Set temp	21
			Time2	Hour	22
				Min	00
				Set temp	16
			Time3	Hour	24
				Min	00
				Set temp	16
			Time4	Hour	24
				Min	00
				Set temp	16
71	75	Weekday	Time1 / On	Hour	07
				Min	00
			Time1 / Off	Hour	09
				Min	00
			Time2 / On	Hour	16
				Min	00
			Time2 / Off	Hour	20
				Min	00
			Time3 / On	Hour	24
				Min	00
			Time3 / Off	Hour	24
				Min	00
			Time4 / On	Hour	24
				Min	00
			Time4 / Off	Hour	24
				Min	00
87	91	Weekend	Time1 / On	Hour	07
				Min	00
			Time1 / Off	Hour	09
				Min	00

			Time2 / On	Hour	16
				Min	00
			Time2 / Off	Hour	20
				Min	00
			Time3 / On	Hour	24
				Min	00
			Time3 / Off	Hour	24
				Min	00
			Time4 / On	Hour	24
				Min	00
			Time4 / Off	Hour	24
	106			Min <b>Dcb finish here if program mode is 00 (5/2 day)</b>	00
103	107	Mon	Time1	Hour	07
				Min	00
				Set temp	21
			Time2	Hour	09
				Min	00
				Set temp	16
			Time3	Hour	16
				Min	00
				Set temp	21
			Time4	Hour	22
				Min	00
				Set temp	16
115	119	Tue	Time1	Hour	07
				Min	00
				Set temp	21
			Time2	Hour	09
				Min	00
				Set temp	16
			Time3	Hour	16
				Min	00
				Set temp	21
			Time4	Hour	22
				Min	00
				Set temp	16
127	131	Wed	Time1	Hour	07
				Min	00
				Set temp	21
			Time2	Hour	09
				Min	00
				Set temp	16
			Time3	Hour	16
				Min	00
				Set temp	21
			Time4	Hour	22
				Min	00
				Set temp	16
139	143	Thu	Time1	Hour	07
				Min	00
				Set temp	21
			Time2	Hour	09
				Min	00

				Set temp	16
			Time3	Hour	16
				Min	00
				Set temp	21
			Time4	Hour	22
				Min	00
				Set temp	16
151	155	Fri	Time1	Hour	07
				Min	00
				Set temp	21
			Time2	Hour	09
				Min	00
				Set temp	16
			Time3	Hour	16
				Min	00
				Set temp	21
			Time4	Hour	22
				Min	00
				Set temp	16
163	167	Sat	Time1	Hour	09
				Min	00
				Set temp	21
			Time2	Hour	22
				Min	00
				Set temp	16
			Time3	Hour	24
				Min	00
				Set temp	16
			Time4	Hour	24
				Min	00
				Set temp	16
175	179	Sun	Time1	Hour	09
				Min	00
				Set temp	21
			Time2	Hour	22
				Min	00
				Set temp	16
			Time3	Hour	24
				Min	00
				Set temp	16
			Time4	Hour	24
				Min	00
				Set temp	16
187	191	Monday	Time1 / On	Hour	07
				Min	00
			Time1 / Off	Hour	09
				Min	00
			Time2 / On	Hour	16
				Min	00
			Time2 / Off	Hour	20
				Min	00
			Time3 / On	Hour	24
				Min	00
			Time3 / Off	Hour	24
				Min	00

			Time4 / On	Hour	24
				Min	00
			Time4 / Off	Hour	24
				Min	00
203	207	Tue	Time1 / On	Hour	07
				Min	00
			Time1 / Off	Hour	09
				Min	00
			Time2 / On	Hour	16
				Min	00
			Time2 / Off	Hour	20
				Min	00
			Time3 / On	Hour	24
				Min	00
			Time3 / Off	Hour	24
				Min	00
			Time4 / On	Hour	24
				Min	00
			Time4 / Off	Hour	24
				Min	00
219	223	Wed	Time1 / On	Hour	07
				Min	00
			Time1 / Off	Hour	09
				Min	00
			Time2 / On	Hour	16
				Min	00
			Time2 / Off	Hour	20
				Min	00
			Time3 / On	Hour	24
				Min	00
			Time3 / Off	Hour	24
				Min	00
		Thu	Time4 / On	Hour	24
				Min	00
			Time4 / Off	Hour	24
				Min	00
235	239		Time1 / On	Hour	07
				Min	00
			Time1 / Off	Hour	09
				Min	00
			Time2 / On	Hour	16
				Min	00
			Time2 / Off	Hour	20
				Min	00
			Time3 / On	Hour	24
				Min	00
			Time3 / Off	Hour	24
				Min	00
			Time4 / On	Hour	24
				Min	00
			Time4 / Off	Hour	24
				Min	00
251	255	Fri	Time1 / On	Hour	07
				Min	00
			Time1 / Off	Hour	09

				Min	00
			Time2 / On	Hour	16
				Min	00
			Time2 / Off	Hour	20
				Min	00
			Time3 / On	Hour	24
				Min	00
			Time3 / Off	Hour	24
				Min	00
			Time4 / On	Hour	24
				Min	00
			Time4 / Off	Hour	24
				Min	00
267	271	Sat	Time1 / On	Hour	07
				Min	00
			Time1 / Off	Hour	09
				Min	00
			Time2 / On	Hour	16
				Min	00
			Time2 / Off	Hour	20
				Min	00
			Time3 / On	Hour	24
				Min	00
			Time3 / Off	Hour	24
				Min	00
		Sun	Time4 / On	Hour	24
				Min	00
			Time4 / Off	Hour	24
				Min	00
283	287		Time1 / On	Hour	07
				Min	00
			Time1 / Off	Hour	09
				Min	00
			Time2 / On	Hour	16
				Min	00
			Time2 / Off	Hour	20
				Min	00
			Time3 / On	Hour	24
				Min	00
			Time3 / Off	Hour	24
				Min	00
			Time4 / On	Hour	24
				Min	00
			Time4 / Off	Hour	24
				Min	00
	302				

6.3 DCB for TM1 WI-FI / WI-FI-RF, read the entire dcb at once

Unique address	Index in DCB	Comments	Default value (dec)
0	0	DCB length in byte, high 8 bit	
	1	DCB length in byte, low 8 bit	
2	2	Vendor ID 00: Heatmiser 01: OEM	00
3	3	Version 10 = VER 1.0	10



4	4	Model (TM1 = 5)		05		
11	5	Address [1, 32], not used		00		
16	6	Program mode 00: 5/2 mode 01: 7 day mode 02: Countdown timer mode		00		
31	7	Away mode 00=normal mode 01=away mode		00		
21	8	On/Off (R/W) 00: off, 01: on		00		
22	9	Key lock (R/W) 00: key unlock, 01: key lock		00		
24	10	Holiday return date, year		Write holiday return date&time to issue a holiday, write one byte 00 to cancel it		
	11	Holiday return date, month				
	12	Holiday return date, day of month				
	13	Holiday return time, hour				
	14	Holiday return time, minute				
24	15	Holiday Enabled 00 = not in holiday mode currently for reading, if write 00 will cancel holiday 01 = currently in holiday mode, writing will not accept 01 value				
26	16	Countdown timer min, high 8 bit		[0,1800] min		
	17	Countdown timer min, low 8 bit High 8 bit ahead when read Low 8 bit ahead when write				
42	18	Current timer state(R/W) Read 00: timer is off now Read 01: timer is on now Write 00, let timer run based on programmed timer period Write 01, override timer to on Write 02, override timer to off		00		
43	19	Year, [00,990				
	20	Month, [1,12]				
	21	Day of month, [1,31]				
	22	Week 1~7 = Mon. ~ Sun				
	23	Hour [0, 23]				
	24	Min [0, 59]				
	25	Sec [0, 59] <b>DCB finish here if program mode is 02 (countdown timer mode)</b>				
71	26	Weekday	Time1 / On	Hour	07	
				Min	00	
				Time1 / Off	Hour	09
				Min	00	
				Time2 / On	Hour	16
				Min	00	
				Time2 / Off	Hour	20
				Min	00	
				Time3 / On	Hour	24
				Min	00	
				Time3 / Off	Hour	24
				Min	00	
				Time4 / On	Hour	24
				Min	00	

			Time4 / Off	Hour	24
				Min	00
87	42	Weekend	Time1 / On	Hour	07
				Min	00
			Time1 / Off	Hour	09
				Min	00
			Time2 / On	Hour	16
				Min	00
			Time2 / Off	Hour	20
				Min	00
			Time3 / On	Hour	24
				Min	00
			Time3 / Off	Hour	24
				Min	00
			Time4 / On	Hour	24
				Min	00
			Time4 / Off	Hour	24
				Min	00
				<b>Dcb finish here if program mode is 00 (5/2 day)</b>	
187	58	Monday	Time1 / On	Hour	07
				Min	00
			Time1 / Off	Hour	09
				Min	00
			Time2 / On	Hour	16
				Min	00
			Time2 / Off	Hour	20
				Min	00
			Time3 / On	Hour	24
				Min	00
			Time3 / Off	Hour	24
				Min	00
			Time4 / On	Hour	24
				Min	00
			Time4 / Off	Hour	24
				Min	00
203	74	Tue	Time1 / On	Hour	07
				Min	00
			Time1 / Off	Hour	09
				Min	00
			Time2 / On	Hour	16
				Min	00
			Time2 / Off	Hour	20
				Min	00
			Time3 / On	Hour	24
				Min	00
			Time3 / Off	Hour	24
				Min	00
			Time4 / On	Hour	24
				Min	00
			Time4 / Off	Hour	24
				Min	00
219	90	Wed	Time1 / On	Hour	07
				Min	00
			Time1 / Off	Hour	09

				Min	00
			Time2 / On	Hour	16
				Min	00
			Time2 / Off	Hour	20
				Min	00
			Time3 / On	Hour	24
				Min	00
			Time3 / Off	Hour	24
				Min	00
			Time4 / On	Hour	24
				Min	00
			Time4 / Off	Hour	24
				Min	00
235	106	Thu	Time1 / On	Hour	07
				Min	00
			Time1 / Off	Hour	09
				Min	00
			Time2 / On	Hour	16
				Min	00
			Time2 / Off	Hour	20
				Min	00
			Time3 / On	Hour	24
				Min	00
			Time3 / Off	Hour	24
				Min	00
		Fri	Time4 / On	Hour	24
				Min	00
			Time4 / Off	Hour	24
				Min	00
251	122		Time1 / On	Hour	07
				Min	00
			Time1 / Off	Hour	09
				Min	00
			Time2 / On	Hour	16
				Min	00
			Time2 / Off	Hour	20
				Min	00
			Time3 / On	Hour	24
				Min	00
			Time3 / Off	Hour	24
				Min	00
			Time4 / On	Hour	24
				Min	00
			Time4 / Off	Hour	24
				Min	00
267	138	Sat	Time1 / On	Hour	07
				Min	00
			Time1 / Off	Hour	09
				Min	00
			Time2 / On	Hour	16
				Min	00
			Time2 / Off	Hour	20
				Min	00
			Time3 / On	Hour	24
				Min	00

			Time3 / Off	Hour	24
				Min	00
			Time4 / On	Hour	24
				Min	00
			Time4 / Off	Hour	24
				Min	00
283	154	Sun	Time1 / On	Hour	07
				Min	00
			Time1 / Off	Hour	09
				Min	00
			Time2 / On	Hour	16
				Min	00
			Time2 / Off	Hour	20
				Min	00
			Time3 / On	Hour	24
				Min	00
			Time3 / Off	Hour	24
				Min	00
			Time4 / On	Hour	24
				Min	00
			Time4 / Off	Hour	24
	169			Min	00

## 7.Muti-Link /NM & multi-zone application protocol

This protocol is used to talk between the multi-link/nm and the iphone/android app for multiple zones application, with this protocol, the app can read from or write to Wi-Fi stats or rs485 stats through multi-link /nm remotely. Port number 8078 for rs485 version and 8082 for Wi-Fi version are used, the app must include both Wi-Fi version and rs485 version stats, they have the same communications protocol but the dcb for Wi-Fi stats and rs485 stats are a little different, the app can identify them from the port number or from the multi-link response.

### 7.1. multi-link /NM dcb (Wi-Fi version or RS485 version)

Unique Address	Contents	Value	Comments
0	DCB length,		variable
1			
2	VID	0=HEATMISER 1=OEM	
3	Version		
4	Model	0x90=rs485 version 0x91=Wi-Fi version	
5	Away 0=home state, 1=away state	Global setting	
6	Time zone	[-12, 12]	
7	GMT correction enable	1=enable, 0=disable	
8	Clock, Year	[00,99]	
	Month	[1, 12]	
	Day of month	[1,31]	
	Week	[1,7]=Mon to Sun	
	Hour	[0, 23]	
	Min	[0, 59]	
	Sec	[0, 59]	
15	Total stats connected	0~32	
16	First Stat ID	1~254 for Wi-Fi or 1~32 for prt	Id of each stat

17		[0,1800] min	Only valid for prthw & tml
19-34		Stat name in 15 char	
...	...	...	...
	Hot water/timer boost in minutes, 2 bytes	1~254 for Wi-Fi or 1~32 for prt	Id of each stat
	stat name, occupy 16 bytes	[0,1800] min	Only valid for prthw & tml
	stat name, occupy 16 bytes	Stat name in 15 char	

## 7.2. Reading DCB from multi-link /nm command

Index	Contents	Value	Comments
0	Frame head	0x98	
1	Frame len, low 8 bit	8	
2	Frame len, high 8 bit	0	
3	PIN, low 8 bit		
4	PIN, high 8 bit		
5	Read from which device	0=read Multi-link dcb Or stat ID	[1,32] for rs485 stat or [1, 254] for Wi-Fi stat
6	16bit crc low 8bit		
7	16bit crc high 8bit		

## 7.3. Reply DCB from multi-link /nm

Byte No	Contents	Value	Comments
0	Frame head	0x99	
1	Frame len, low 8 bit		
2	Frame len, high 8 bit		
3	Frame number	Increment by 1 for each write	[0, 255]
4	Read from which device	0=read Multi-link dcb Or Stat ID	[1,32] for rs485 stat or [1,254] for Wi-Fi stat
5	Statistics	Communication rate, [0~100] 0=offline	If read from Multi-link and statistics=0 means pin error
...	Contents of dcb	If statistics=0, no dcb will be sent	
	16bit crc low 8bit		
	16bit crc high 8bit		

## 7.4 Write to the multi-link /nm to let it change stat/ multi-link /nm parameter (multi-link will handle these commands and forward them if they need to be sent to stats)

Byte No.	Contents	Value	Comments
0	Frame head	0xa8	
1	Frame len, low 8 bit		
2	Frame len, high 8 bit		
3	Pin, low 8 bit		
4	Pin, high 8 bit		
5	Total writing command numbers	Mostly 32	
6	Stat ID	[1,32]=rs485 stats or [1,254]=Wi-Fi stat or 0=Multi-link	
7	Write to which address, low 8 bit	Stat dcb write address or	
8	Write to which address, high 8 bit	Multi-link dcb write	

		address	
9	Number of bytes to be written		
10	Contents to be written		
	Stat ID	[1,32]=rs485 stats or [1,254]=Wi-Fi stat or 0=Multi-link	
	Write to which address, low 8 bit	Stat dcb write address or Multi-link dcb write address	
	Write to which address, high 8 bit		
	Number of bytes to be written		
	Contents to be written		
...	...		
	16bit crc low 8 bit		
	16bit crc high 8 bit		

If this frame is used to set multi-link /nm parameters, will reply multi-link /nm dcb

If this frame is used to set the dcb of the same one stat, will return the dcb of this stat

If this frame contains mixed commands that will be sent to several different stats, it will reply with a summary frame.

Byte No	Contents	Value	Comments
0	Frame head	0x99	
1	Frame len, low 8 bit	7	
2	Frame len, high 8 bit	0	
3	which device	0=0xff, means mixed commands	
4	Error or not	100=no error in the frame 0=error in this frame	
5	16bit crc low 8bit		
6	16bit crc high 8bit		

For example if we want to set Away to stat 100, 101, and set key lock to stat 102, 103, and set setback to 104, 105, and temp hold 28 degrees for 5 hours to stat 106, 107, we should use below frame

```

Cmd[0] = 0xa8;           // frame head
Cmd[1] = 60;             // frame length, crc included
Cmd[2] = 0;
Cmd[3] = 0xd4;           // pin, 1234
Cmd[4] = 0x02;
Cmd[5] = 10;             // Number of commands in this frame
// 1st command
Cmd[6] = 100;            // ID of the first stat
Cmd[7] = 31;             // AWAY command write address for Wi-Fi stat
Cmd[8] = 0;
Cmd[9] = 1;              // number of bytes to be written
Cmd[10] = 1;             // set AWAY enable
// 2nd command
Cmd[11] = 101;           // ID of the second stat
Cmd[12] = 31;            // AWAY command write address for Wi-Fi stat
Cmd[13] = 0;
Cmd[14] = 1;             // number of bytes to be written
Cmd[15] = 1;            // set AWAY enable
// 3rd command
Cmd[16] = 102;           // ID of the third stat
Cmd[17] = 22;            // KEYLOCK command write address for Wi-Fi stat
Cmd[18] = 0;
Cmd[19] = 1;             // number of bytes to be written
Cmd[20] = 1;            // set KEYLOCK on
// 4th command

```

```

Cmd[21] = 103;           // ID of the 4th stat
Cmd[22] = 22;           // KEYLOCK command write address for Wi-Fi stat
Cmd[23] = 0;
Cmd[24] = 1;           // number of bytes to be written
Cmd[25] = 1;           // set KEYLOCK on
// 5th command
Cmd[26] = 104;          // ID of the 5th stat
Cmd[27] = 23;           // SETBACK command write address for Wi-Fi stat
Cmd[28] = 0;
Cmd[29] = 1;           // number of bytes to be written
Cmd[30] = 1;           // set SETBACK on
// 6th command
Cmd[31] = 105;          // ID of the 6th stat
Cmd[32] = 23;           // SETBACK command write address for Wi-Fi stat
Cmd[33] = 0;
Cmd[34] = 1;           // number of bytes to be written
Cmd[35] = 1;           // set SETBACK on
// 7th command
Cmd[36] = 106;          // ID of the 7th stat
Cmd[37] = 18;           // SET TEMP command write address for Wi-Fi stat
Cmd[38] = 0;
Cmd[39] = 1;           // number of bytes to be written
Cmd[40] = 28;           // set to 28 degree
// 8th command
Cmd[41] = 106;          // ID of the 7th stat
Cmd[42] = 32;           // TEMP HOLD command write address for Wi-Fi stat
Cmd[43] = 0;
Cmd[44] = 2;           // number of bytes to be written
Cmd[45] = (unsigned char)300; // 300 min low byte
Cmd[46] = (unsigned char)(300/256); // 300 min high byte
// 9th command
Cmd[47] = 107;          // ID of the 8th stat
Cmd[48] = 18;           // SET TEMP command write address for Wi-Fi stat
Cmd[49] = 0;
Cmd[50] = 1;           // number of bytes to be written
Cmd[51] = 28;           // set to 28 degree
// 10th command
Cmd[52] = 107;          // ID of the 8th stat
Cmd[53] = 32;           // TEMP HOLD command write address for Wi-Fi stat
Cmd[54] = 0;
Cmd[55] = 2;           // number of bytes to be written
Cmd[56] = (unsigned char)300; // 300 min low byte
Cmd[57] = (unsigned char)(300/256); // 300 min high byte

Cmd[58] = crclow;
Cmd[59] = crchi;

```

After the multi-link /nm received this frame, it will split it into 10 commands and dispatch them to the corresponding stats, of course it will take a long time to send all of them to their destination so they will be saved in a command buffer first, then a reply will be sent to the app with the frame that is shown in the previous table.

Another procedure will send these 10 commands one by one. In this way the app can just send multiple commands in a single frame to the multi-link /nm and let the multi-link /nm send the data to the stats.

You can of course send a single command in this network frame.

Example to set multi-link/nm parameters (AWAY, TIME ZONE & GMT)

Byte No.	Contents	Value	Comments
----------	----------	-------	----------

0	Frame head	0xa8	
1	Frame len, low 8 bit		
2	Frame len, high 8 bit		
3	Pin, low 8 bit		
4	Pin, high 8 bit		
5	Total writing item numbers	Mostly 32	
6	Write to which device	0=Multi-link/nm, fixed to 0	
7	Write to which address (low byte)	5=Away 6=Time zone 7=GMT correction enable	
8	Write to which address (high byte)	0	
9	Number of bytes to be written	1	
10	Away (global setting) Time zone GMT correction enable	1=away, 0=home [-12, 12] 1=Enable, 0=Disable	
11	16bit crc low 8 bit		
12	16bit crc high 8 bit		

When multi-link/nm gets an AWAY command, it will generate AWAY commands for each of the connected stats.

multi-link /nm will reply with a dcb for this frame.

Example to set clock for multi-link /nm

Byte No.	Contents	Value	Comments
0	Frame head	0xa8	
1	Frame len, low 8 bit	19	
2	Frame len, high 8 bit	0	
3	Pin, low 8 bit		
4	Pin, high 8 bit		
5	Total writing item numbers	1	
6	Write to which device	0= multi-link /nm, fixed to 0	
7	Write to which address, (low byte)	8=clock	
8	Write to which address, (hi byte)	0	
9	Number of bytes to be written	7	
10	Year	[0,99]	
11	Month	[1,12]	
12	Day of month	[1, 31]	
13	Week	[1,7]=Mon to Sun	
14	Hour	[0,23]	
15	Min	[0,59]	
16	Sec	[0,59]	
17	16bit crc low 8 bit		
18	16bit crc high 8 bit		

When multi-link /nm get this frame, it will generate CLOCK commands for each connected stat and then reply with a dcb to the app.

Example to set hot water/timer boost for 300 min to stat 100 & 101 (only valid for prthw and tm1 in 5/2 or 7 day program mode)

Byte No.	Contents	Value	Comments
0	Frame head	0xa8	
1	Frame len, low 8 bit	22	
2	Frame len, high 8 bit	0	
3	Pin, low 8 bit		
4	Pin, high 8 bit		
5	Number of commands to be set	2	



6	Write to which device	0=i multi-link box/nm	
7	Write to which address (low byte)	17=hw/timer boost in min	
8	Write to which address (high byte)	0	
9	Number of bytes to be written	2	
10	to which stat (ID)	[1,32] or [1,254]	
11	Boost min low byte	0x2c	
12	Boost min hi byte	0x01	
13	Write to which device	0= multi-link /nm	
14	Write to which address (low byte)	17=hw/timer boost in min	
15	Write to which address (high byte)	0	
16	Number of bytes to be written	2	
17	to which stat (ID)	[1,32] or [1,254]	
18	Boost min low byte	0x2c	
19	Boost min hi byte	0x01	
20	16bit crc low 8 bit		
21	16bit crc high 8 bit		

NB this frame is a little different from the others since we add item no.17 to identify which stat (ID) will be set with the boost command. When the multi-link /nm receives' this frame, it will split it into 2 individual commands and send them to their destination stats and then reply with a dcb to the app.

Example to set zone name for example to stat 100 & 101

Byte No.	Contents	Value	Comments
0	Frame head	0xa8	
1	Frame len, low 8 bit	50	
2	Frame len, high 8 bit	0	
3	Pin, low 8 bit		
4	Pin, high 8 bit		
5	Number of commands to be set	2	
6	Write to which device	0= multi-link /nm	
7	Write to which address (low byte)	19=zone name	
8	Write to which address (hi byte)	0	
9	Number of bytes to be written	16	
10	Write to which stat (ID)	100	
11-26	Zone name, 16 bytes	None used bytes must be fill with zone	
27	Write to which device	0= multi-link /nm	
28	Write to which address (low byte)	19=zone name	
29	Write to which address (hi byte)	0	
30	Number of bytes to be written	16	
31	Write to which stat (ID)	101	
32-47	Zone name, 16 bytes	None used bytes must be fill with zone	
48	16bit crc low 8 bit		
49	16bit crc high 8 bit		

When the multi-link /nm receives this frame, it will change the zone name for stat with ID=100 & 101 at the same time, then reply with a dcb to the app

#### 7.5. Reading/writing history data from/to multi-link /nm command

Index	Contents	Value	Comments
0	Frame head	0x95	
1	Frame len, low 8 bit		
2	Frame len, high 8 bit		
3	PIN, low 8 bit		
4	PIN, high 8 bit		
5	which device	ID of the stat	[1,32] for rs485 stat or

			[1, 254] for Wi-Fi stat
6	Function code	00=read 01=erase history 02=erase hours run 03=erase both history and hours run	
7	16bit crc low 8bit		
8	16bit crc high 8bit		

#### 7.6. reply logging data from Multi-link/nm command

Index	Contents	Value	Comments
0	Frame head	0x96	
1	Frame len, low 8 bit		6=no logging data
2	Frame len, high 8 bit		
3	Logging data from which device	ID of the stat	[1,32] for rs485 stat or [1, 254] for Wi-Fi stat
4	Login data contents		698 bytes
	16bit crc low 8bit		
	16bit crc high 8bit		

#### 7.7 Logging data format

Index	Contents	Value	comments
0	Stat ID		
1	Mon logging	History data, in C format, point/15 min, 96 bytes/day, each day occupy 96 bytes	
97	Tue logging		
193	Wed logging		
289	Thu logging		
385	Fri logging		
481	Sat logging		
577	Sun logging		
673	Hours run year	[00, 99]	
674	Jan hours run	[0, 744]	
676	Feb hours run		
678	Mar hours run		
680	Apr hours run		
682	May hours run		
684	Jun hours run		
686	July hours run		
688	Aug hours run		
690	Sep hours run		
692	Oct hours run		
694	Nov hours run		
696	Dec hours run		

#### 7.8 Read from multi-link /nm to get the summary information of all stats

Inquire command sent from the app

Index	Contents	Value	Comments
0	Frame head	0x95	
1	Frame len, low 8 bit	9	
2	Frame len, high 8 bit		
3	PIN, low 8 bit		
4	PIN, high 8 bit		
5	which device	ID of the multi-link	Fixed to 0, multi-link
6	Function code	00=read	Fixed to 0
7	16bit crc low 8bit		

8	16bit crc high 8bit		
---	---------------------	--	--

#### Reply from Multi-link/nm

Index	Contents	Value	Comments
0	Frame head	0x96	
1	Frame len, low 8 bit		6=no stat data
2	Frame len, high 8 bit		
3	ID	ID of the multi-link	Fixed to 0
4	All stats summary information	See below table	Dynamic length
	16bit crc low 8bit		
	16bit crc high 8bit		

#### Contents of All stats summary information

Index	Contents	Value	Comments
0	Total number of stats	0 to 32, 1 byte	
1	ID of the first stat	[1,32] or [1, 254] 1 byte	Comms number of rs485 stats Ip last byte of Wi-Fi stat
2-17	Zone name	16 bytes	Last one byte is fixed to 0 as the terminal of this string
18	Model + program option	0xff=offline Bit3:0=Model Bit7:4=program mode 1 byte	If offline, will no zone temp, set temp and status segments
19	Zone temp, low byte	1 byte	((high byte)x256+(low byte))/10 = room temp
20	Zone temp, high byte	1 byte	
21	Set temp if in heating mode Frost temp if in frost mode	1 byte	
22	Status	1 byte	Bit0=Lock status Bit1=Heating status Bit2=Frost status Bit3=HW/TIMER status Bit4=Temperature format Bit5=Summer status Bit6=Holiday status Bit7=Away
23	ID of the second stat	[1,32] or [1, 254] 1 byte	Comms number of rs485 stats Ip last byte of Wi-Fi stat
...	...	...	...

More often than not the contents have  $1+22 \times 32=705$  bytes

#### Reply from multi-link /nm for the writing command

When a write command is received by the multi-link /nm, if this command is used to set stat parameters, multi-link /nm needs to forward them to the corresponding Wi-Fi/prt stat, if it is a Wi-Fi stat, it will reply with it's dcb to the multi-link right away, when the multi-link gets this reply, it should forward this frame to the app so the app can show this change immediately; If it is a rs485 stat forwarded by nm, after writing to the stat, nm should read the stat dcb right away and when nm gets the reply, it should forward this frame to the app so the app can show this change immediately.

#### 8. Address table, this address are the comms number used in RS485 network

Equipment	Address
V3 Stats	[1, 32]
HTS (humidity sensor)	[41, 72]
OTS (outside temp sensor)	81
MC V3	0x81 (129)

TR2 V3	0x82
NetMonitor V3	0x83
TFT V3	0x81
UHC	0x91-0xA0 [145, 160]

## 9. Timing considerations when reading and writing on the RS485 network

We should wait for a certain period of time for the reply when reading and writing

### Reading.

After sending a read request the master node must enter into a wait state with a 1000ms timeout.

Normally a complete reply will be received within 1000ms (depending on frame length),

For example if the whole reply is received within say 700ms, the master must wait another 100 mS before sending the next frame, this allows the bus time to recover.

If any stat only receives a part of a frame, it needs at least 20 mS. before it will timeout and reset the communications).

After 1000mS the next frame can be sent

After sending a write request the master node must enter into a wait state with a 1000ms timeout.

Normally a reply will be received within 1000ms but only after the thermostat has completely handled the frame. Normally processing only takes a few hundred mS.

If the thermostat response is incorrect the master can re send the command up to 5 times to re-establish the connection.

The Broadcast address is FF hex and can only be used for writing data, no reply will be sent by the thermostat please allow at least 1000mS between broadcasts and repeat each broadcast at least 5 times

# APPENDIX A

## CRC calculation code

```
static const unsigned char CRC16_LookupHigh[16] = {
    0x00, 0x10, 0x20, 0x30, 0x40, 0x50, 0x60, 0x70,
    0x81, 0x91, 0xA1, 0xB1, 0xC1, 0xD1, 0xE1, 0xF1
};
static const unsigned char CRC16_LookupLow[16] = {
    0x00, 0x21, 0x42, 0x63, 0x84, 0xA5, 0xC6, 0xE7,
    0x08, 0x29, 0x4A, 0x6B, 0x8C, 0xAD, 0xCE, 0xEF
};

unsigned char CRC16_High, CRC16_Low;

void CRC16_Update4Bits(unsigned char val )
{
    unsigned char t;

    // Step one, extract the Most significant 4 bits of the CRC register
    t = CRC16_High >> 4;

    // XOR in the Message Data into the extracted bits
    t = t ^ val;

    // Shift the CRC Register left 4 bits
    CRC16_High = (CRC16_High << 4) | (CRC16_Low >> 4);
    CRC16_Low = CRC16_Low << 4;

    // Do the table lookups and XOR the result into the CRC Tables
    CRC16_High = CRC16_High ^ CRC16_LookupHigh[t];
    CRC16_Low = CRC16_Low ^ CRC16_LookupLow[t];
}

/*
 * Process one Message Byte to update the current CRC Value
 */
void CRC16_Update(unsigned char val )
{
    CRC16_Update4Bits( val >> 4 ); // High nibble first
    CRC16_Update4Bits( val & 0x0f ); // Low nibble
}

void CRC16(char *buf, unsigned int len, unsigned char *low, unsigned char *hi)
{
    CRC16_High = 0xff;
    CRC16_Low = 0xff;
    __RESET_WATCHDOG();
    while(len--)
    {
        CRC16_Update(*buf++);
    }
    *low = CRC16_Low;
    *hi = CRC16_High;
}
```