

User manual

# WIRELESS DC TOOL COM Protocol MODBUS BM/BMT series



EN

60427-05/23





# INDEX

1. Overview and Communication Specifications .....	4
1.1 Overview .....	4
2. Basic Structure of Communication Protocol MODBUS TCP .....	5
2.1 Modbus TCP Frame Format .....	5
3. Function code & message details .....	6
3.1 < READ > Function code 03 & 04 details.....	6
3.2 < WRITE > Function code 06 : writing single parameter .....	9
3.3 < WRITE > Function code 16 : writing multiple parameters.....	11
3.4 Auto data output – data description .....	13
3.5 < REQUEST > Function code 17 : Slave device information.....	14
3.6 Error code for abnormal response .....	16
4. Appendix A .....	17
4.1 Parameter factory setting, Address and Function code details for BM .....	17
4.2 Parameter factory setting, Address and Function code details for BMT .....	17

## 1. Overview and Communication Specifications

### 1.1 Overview

BM(T) are capable of connecting to the host controller (Handy Loader, HMI, PLC, PC, etc.) through Wifi & ethernet, allowing the user to use such functions as parameter change and data monitoring.

- Direct connection through an access point



AP : WiFi Access Point or Router 2.4GHz or 5GHz (not included)

Each screwdriver should have a different IP address.

- Connection through ParaMonPro X controller



ParaMonPro X is used as a Modbus gateway to communicate through Ethernet LAN connection with connected screwdrivers.

Screwdriver identification(up to 8 tools) is made with slave address in Modbus frame.

## 2. Basic Structure of Communication Protocol MODBUS TCP

### 2.1 Modbus TCP Frame Format

Name	Length(byte)	Function
Transaction Identifier	2	For synchronization between messages of server & client
Protocol Identifier	2	Zero for Modbus/TCP
Length Field	2	Number of remaining bytes in this frame
Unit Identifier	1	Slave Address
Function	1	Function codes as in other variants
Data	n	Data as response or commands

Format of Modbus/TCP frame is described in the below figure.

**MODBUS TCP Frame Structure**



- byte 0 ~ 1: transaction ID (Transaction Identification)

This means the sequence number of queries and responses. While TCP operates as a master, it is incremented by one in every query (It doesn't matter if this field is set to 0x0000).

- byte 2 ~ 3: protocol ID (Protocol Identification)

This means the protocol identification and the value is fixed as 0x0000 for Modbus/TCP

- byte 4 ~ 5: length

The value of this means the number of bytes from next byte of length field to the end of the frame.

- byte 6: unit ID(Slave Address)  
set to 0 for direct COM with tool (1 tool per IP address)  
or tool number (0 to 7) if COM goes through ParaMonPro X in gateway mode.
- byte 7: FC (Function Code)
- byte 8~ : data depending on function code

### 3. Function code & message details

Function code	Description	Remark
03 (0x03)	<b>Read</b> Holding Register	16bit data (Integer) ex) parameter
04 (0x04)	<b>Read</b> Input Register	16bit data ex) monitoring data
06 (0x06)	<b>Write</b> Single Register	16bit integer format ex) parameter data
16 (0x16)	<b>Write</b> Multiple Registers	16bit integer format ex) parameter data
17 (0x11)	<b>Request</b> Slave ID	

#### 3.1 < READ > Function code 03 & 04 details

Function code 03 is used to read the register as like parameters (address 1 to 725)

Function code 04 is used to read the register as like alarm & monitoring data (address 3100 to 3399).

The only integer number is allowed.

[Query ( Request) ]

Slave address	Function Code	Start address High	Start address Low	No of address High	No of address Low
---------------	---------------	--------------------	-------------------	--------------------	-------------------

Function code consist of one byte. But start address and number of address are consisted by 2 bytes with 4 digits of hexadecimal, starting with first 2 digits for high number, second 2 digits for low number.

[OK Response ]

Slave address	Function Code	No of byte	Data #1 High	Data #1 Low	... x n data
---------------	---------------	------------	--------------	-------------	--------------

The number of data is consisted by 2 bytes with 4 digits of hexadecimal. So the total number of data is equal to 1/2 of the number of byte.

[Abnormal Response]

Slave address	Function code 0x080	Error code
---------------	------------------------	------------

By adding 0x080 to the function code, it response any abnormal or wrong message

### Example message of “ 03 “ for parameter data

To read the data of parameter 1 and 2, which is torque & speed value of Preset #1

( 03 : Query)

Slave address	03	00	02	00	02
---------------	----	----	----	----	----

Start address : 2 ( 0002h )

Number of address : 2 (0002h)

Read data of two addresses starting from address #2

( 03 : Response )

Slave address	Function Code	No of byte	Data #1 High	Data #1 Low	... x n data
---------------	---------------	------------	--------------	-------------	--------------

Slave address	03	04	01	0F	03	21
---------------	----	----	----	----	----	----

Data value of 1<sup>st</sup> address : 010F (hex) = 271 (dec) ← torque value of Preset #1

Date value of 2<sup>nd</sup> address : 03E8(hex) = 1000 (dec) ← Torque limit value 10.00%

[ Address for parameters ]

Refer the appendix A for all address details for parameters

1 – 725 : Parameter address

( 1001 – 1724 : Parameter value range MIN )

( 2001 – 2724 : Parameter value range MAX )

### Example message of “ 04 ” for fastening data

To read the fastening data which is renewed by the event such as start, stop, F/R rotation...

( 04 : Query )

Slave address	04	0c	84	00	02
---------------	----	----	----	----	----

Start address : 3204 ( 0c84h )

Number of address : 2 ( 0002h )

Read data of two addresses starting from address #3204

( 04 : Response )

Slave address	Function Code	No of byte	Data #1 High	Data #1 Low	... x n data
---------------	---------------	------------	--------------	-------------	--------------

Slave address	04	04	01	0F	03	21
---------------	----	----	----	----	----	----

Data value of 1<sup>st</sup> address : 010Fh = 2.71 (dec) ← Converted torque value

Data value of 2<sup>nd</sup> address : 03E8h = 1000 (dec) ← Target speed

[ Address for monitoring ]

Refer the appendix A for all address details for monitoring

3100 – 3101 : Alarm data

3200 – 3213 : Data updated by event ( Start, F/L, Preset change, Torque up )

3300 – 3313 : Real-time data



### 3.2 < WRITE > Function code 06 : writing single parameter

Function code 06 is used to WRITE the parameter value in each register. The only integer number is allowed.

( Query )

Slave address	Function Code	Address High	Address Low	Data High	Data Low

( OK Response )

Slave address	Function Code	Address High	Address Low	Date High	Data Low

It provides the echo response on the query (request) after writing data in register.

- Refer the appendix A for all address details for writing.

4000 – 4005 : remote inputs (preset selection, remote start, driver lock, F/L, alarm reset)

### Example message of “ 06 “ for Real-time monitoring data Auto output → ON

To read the renewed fastening data automatically and continuously, repeat the same query as below in every 10 seconds.

( 06 : Query)

Slave address	06	0F	A2	00	01
---------------	----	----	----	----	----

Address : 4002 ( 0FA2h )

Data : 1 ( 0001h )

( 06 : Response )

It response the same as the query

01	06	0F	A2	00	01
----	----	----	----	----	----

Once it write “ 1 “ on the address 4002, the real-time monitoring data is responded with the function code “04” automatically when there is the event as like torque-up, F/R change, preset # change, etc.

Repeat the same query in every 10 seconds to keep the data output.

### 3.3 < WRITE > Function code 16 : writing multiple parameters

Implemented in firmware :

- BM v1.31.4\_20230210 and newer
- BMT v1.41.5\_20230210 and newer

Function code 16 is used to WRITE several parameter values in multiple register. The only integer number is allowed.

( Query )

Slave address	Function Code	Starting Address High	Starting Address Low	Quantity register High	Quantity Register Low	Byte count	Data #1 High	Data #1 Low	...Data # n ...
---------------	---------------	-----------------------	----------------------	------------------------	-----------------------	------------	--------------	-------------	-----------------

( OK Response )

Slave address	Function Code	Starting Address High	Starting Address Low	Quantity register High	Quantity Register Low
---------------	---------------	-----------------------	----------------------	------------------------	-----------------------

It provides the echo response on the query (request) after writing data in register.

This function can be used with :

- parameters registers 1 to 874
- remote control registers 4000 to 4008

Max number of registers to be written at once: 100.

**Example message of “ 16 “ for writing preset#1 parameters (15 registers)**

(16 : Query)

00 01 00 00 00 25 00 10 00 01 00 0F 1E 00 00 00 65 05 DC 00 00 00 00 00 00 00 00 01 51 00 00  
00 00 00 00 00 28 00 3C 00 46 00 60

**Modbus frame :**

- Transaction ID 0x0001/Protocol ID 0x0000/Length Field 0x0025(37bytes)/Unit ID 0x00
- **Function code 16 :0x10**
- **Starting Address : 0x0001**
- **Quantity of Register : 0x000F (15)**
- **Byte count : 0x001E (30 = Register x2)**
- Data # : 15 registers datas

(16 : Response )

00 01 00 00 00 06 00 10 00 01 00 0F

**Modbus frame :**

- Transaction ID 0x0001/Protocol ID 0x0000/Length Field 0x0006(6bytes)/Unit ID 0x00
- **Function code 16 :0x10**
- **Starting Address : 0x0001**
- **Quantity of Register : 0x000F (15)**

### 3.4 Auto data output – data description

Auto data output is same data format as the answer to a read request (code 04) of 14 register addresses 3200 to 3213 but sent spontaneously without read request when a new event occurs

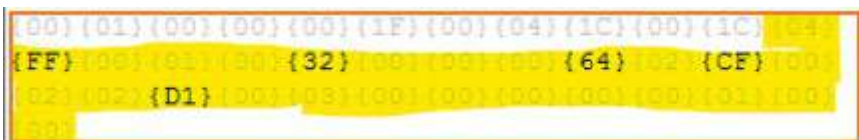
Data description :

Data updated on events (Start, F/L, Preset, Torque up)	Event count no. ( 1- 65,535 )	3200	Read : 0x04	
	Fastening time (ms)	3201	Read : 0x04	
	Preset no.	3202	Read : 0x04	
	Target torque ( * x 100 )	3203	Read : 0x04	[value: 4321] -> [mean: 43.21]
	Converted torque ( * x 100 )	3204	Read : 0x04	
	Target speed (rpm)	3205	Read : 0x04	
	A1 (degree)	3206	Read : 0x04	
	A2 (degree)	3207	Read : 0x04	
	A3 (degree)	3208	Read : 0x04	
	Screw count value	3209	Read : 0x04	
	Error	3210	Read : 0x04	
	Forward / Loosening	3211	Read : 0x04	0: Forward, 1: Loosening
	Status	3212	Read : 0x04	Other = 0, Fastening complete = 1, Fastening NG (E330,332,333,334,335,336,337,338,339)= 2, F/L change = 3, Preset change = 4, Alarm reset = 5, Error(except fastening NG) = 6 Barcode = 7 Screw count-1 = 8
	Snug torque angle (degree)	3213	Read : 0x04	

Event generating an auto data output for any of the following status :

- 0 = Other
- 1 = Fastening complete
- 2 = Fastening NG (E330,332,333,334,335,336,337,338,339)
- 3 = F/L change
- 4 = Preset change
- 5 = Alarm reset
- 6 = Error(except fastening NG)
- 7 = Barcode (not implemented)
- 8 = Screw count-1

Example of auto data output frame:



Auto data frame explanation:

- 00 01 : transmission ID
- 00 00 : protocol ID
- 00 1F : length in bytes (31 in bytes)
- 00 : unit
- 04 : modbus function code (read)
- 1C: number of bytes (28 in dec)
- 00** .... : data registers 3200 to 3213

### 3.5 < REQUEST > Function code 17 : Slave device information

Function code 17 is used to read the slave device information about ID no, controller model, screwdriver model, serial no and firmware version.



#### Information

This request can be used at communication open to monitor controller & screwdriver connected and also as a keep alive polling with tool to check connection.

( Query )

Slave address	Function Code (17)
---------------	--------------------

( Response )

Slave address	Function Code	No. of byte	Driver ID High	Driver ID Low	Controller model High	Controller model Low	Driver model High	Driver model Low	FW Ver. High	FW Ver. Low	S/N 5	S/N 4	S/N 3	S/N 2	S/N 1

#### Driver ID High/Low :

by default 0x00 0x01

#### Tool controller list

Value	BM tool (refer SN3)	BMT tool (refer SN3)
1	BM	BMT

#### Firmware version (Controller)

Value High/Low	BM or BMT tool firmware
0x0D 0x02 1302	v1.30.2

## Driver model list

Value	BM tool model	BMT tool model
0	Unknown	Unknown
1	BM(H)3201	Unknown
2	BM(H)3202	Unknown
3	BM(H)3204	BMT(H)3204
4	BM(H)3206	BMT(H)3206
5	BM(H)3211	BMT(H)3211
6	BM(H)3216	BMT(H)3216
7	BM(H)3224	BMT(H)3224
8	BM(H)3236	BMT(H)3236
9	BM(H)3245	BMT(H)3245
10	BM(H)3264	BMT(H)3264

## Serial no (driver serial number)

SN example : 2011100077

SN5	SN4	SN3	SN2	SN1
Product number low	Product number high	Screwdriver series	Month of production	Year of production
0x4D = 77	0x00= 00	0x0A= 10	0x0B= 11	0x14= 20

SN3 details:

- 01. MD(V2)
- 02. AD
- 10. BM
- 15. MDT and MDT+
- 19. BMT

### 3.6 Error code for abnormal response

If there are wrong function code or communication failure by protocol ( parity, LRC, CRC..etc.), there will be no response. The master will show “ TIME OUT ” error.

If the query contains wrong function code or address, the function code + 0x80 will be responded together with the following error code in data registry.

Error code	Description
0x01	No defined function code or wrong function code
0x02	Wrong address or no existing address
0x03	Data length over the capacity
0x07	Wrong CRC value in query
0x0C	Over the number of byte
0x0E	Range of data is not available

#### Example message

To read the 5 parameter data starting from 564 to 568

( 01 : Query)

Slave address	01	02	34	00	05
---------------	----	----	----	----	----

Start address : 564 (0234h), Number of address : 5 (0005h)

Function code 01 is not defined. → Function code error

Parameter from 564 to 568 are not existing. → No existing address

( 01 + 80 : Response )

Slave address	81	02
---------------	----	----

Function code (01) + 0x80 = 81

Error code for wrong data address = 02



## 4. Appendix A

### 4.1 Parameter factory setting, Address and Function code details for BM

Please refer 60451 datasheet

**DOGA®**  
ASSEMBLY TECHNOLOGY

**MODBUS - Address and Function code details**

**BM series**

DOC.60451-8021

**IMPORTANT**

This document is an extension of Modbus manual 0427.

**Address and function codes details BM**

Appendix A: Parameter factory setting, Address and Function code details for BM 202102012

Param #	Parameter	Address	Factory setting	Address Min Value	Address Max Value	Function code
1	Strategy 0: TOAC - 1: ACTM	1	0	1001	1001	Read: 0x03 Write: 0x06
	Torque	2	Auto	1002	1002	Read: 0x03 Write: 0x06
	Torque minimum (%)	3	0	1003	1003	Read: 0x03 Write: 0x06
	Target angle(degrees)	4	0	1004	1004	Read: 0x03 Write: 0x06
	Min angle(degrees)	5	0	1005	1005	Read: 0x03 Write: 0x06
	Max angle(degrees)	6	0	1006	1006	Read: 0x03 Write: 0x06
	Stop torque	7	0	1007	1007	Read: 0x03 Write: 0x06
	Speed (rpm)	8	Auto	1008	1008	Read: 0x03 Write: 0x06
	Max locking angle(degrees)	9	0	1009	1009	Read: 0x03 Write: 0x06
	Free locking speed(rpm)	10	0	1010	1010	Read: 0x03 Write: 0x06
	Self start 1: 30Hz	11	0	1011	1011	Read: 0x03 Write: 0x06
	Starting point (Hz) 10-40	12	Auto	1012	1012	Read: 0x03 Write: 0x06
	Torque rising adaptive (0-200)	13	0	1013	2013	Read: 0x03 Write: 0x06
	Torque up speed(rpm) (0-50% of max)	14	Auto	1014	2014	Read: 0x03 Write: 0x06
	Torque compensation (%) 0-100	15	100	1015	2015	Read: 0x03 Write: 0x06
	TCUAM: ACTM	16	0	1016	2016	Read: 0x03 Write: 0x06
2	Torque	17	Auto	1017	2017	Read: 0x03 Write: 0x06
	Torque minimum (%)	18	0	1018	2018	Read: 0x03 Write: 0x06
	Target angle(degrees)	19	0	1019	2019	Read: 0x03 Write: 0x06
	Min angle(degrees)	20	0	1020	2020	Read: 0x03 Write: 0x06

www.dogasassembly.com

### 4.2 Parameter factory setting, Address and Function code details for BMT

Please refer 60452 datasheet

**DOGA®**  
ASSEMBLY TECHNOLOGY

**MODBUS - Address and Function code details**

**BMT series**

DOC.60452-8021

**IMPORTANT**

This document is an extension of Modbus manual 0427.

**Address and function codes details BMT**

Appendix A: Parameter factory setting, Address and Function code details for BMT 202102012

Param #	Parameter	Address	Factory setting	Address Min Value	Address Max Value	Function code
1	Strategy 0: TOAC - 1: ACTM	1	0	1001	1001	Read: 0x03 Write: 0x06
	Torque	2	Auto	1002	1002	Read: 0x03 Write: 0x06
	Torque minimum (%)	3	0	1003	1003	Read: 0x03 Write: 0x06
	Target angle(degrees)	4	0	1004	1004	Read: 0x03 Write: 0x06
	Min angle(degrees)	5	0	1005	1005	Read: 0x03 Write: 0x06
	Max angle(degrees)	6	0	1006	1006	Read: 0x03 Write: 0x06
	Stop torque	7	0	1007	1007	Read: 0x03 Write: 0x06
	Speed (rpm)	8	Auto	1008	1008	Read: 0x03 Write: 0x06
	Max locking angle(degrees)	9	0	1009	1009	Read: 0x03 Write: 0x06
	Free locking speed(rpm)	10	0	1010	1010	Read: 0x03 Write: 0x06
	Self start 1: 30Hz	11	0	1011	1011	Read: 0x03 Write: 0x06
	Starting point (Hz) 10-40	12	Auto	1012	1012	Read: 0x03 Write: 0x06
	Torque rising adaptive (0-200)	13	0	1013	2013	Read: 0x03 Write: 0x06
	Torque up speed(rpm) (0-50% of max)	14	Auto	1014	2014	Read: 0x03 Write: 0x06
	Torque compensation (%) 0-100	15	100	1015	2015	Read: 0x03 Write: 0x06
	TCUAM: ACTM	16	0	1016	2016	Read: 0x03 Write: 0x06
2	Torque	17	Auto	1017	2017	Read: 0x03 Write: 0x06
	Torque minimum (%)	18	0	1018	2018	Read: 0x03 Write: 0x06
	Target angle(degrees)	19	0	1019	2019	Read: 0x03 Write: 0x06
	Min angle(degrees)	20	0	1020	2020	Read: 0x03 Write: 0x06

www.dogasassembly.com





