

1. 通訊協定：
採用 MODBUS 通訊協定，電腦連接超過 30 台則需訊號擴大器(Repeater)。
2. 傳送模式：
RTU MODE。
3. 通訊方式：
RS485 半雙工方式(Half-Duplex)。
4. MODBUS 命令結構：
 - 4.1. 基本命令格式：均為 16 進制。

START OF FRAME	ADDRESS FIELD	FUNCTION CODE	DATA FIELD	ERROR CHECK	END OF FRAME
----------------	---------------	---------------	------------	-------------	--------------

START OF FRAME : 至少 4 個字元的時間沒有傳送資料。

ADDRESS FIELD : 欲讀取或控制的位址(範圍 1~255)。
廣播方式：Address 0，只對 Function Code：06H 有效且不回應訊息。

FUNCTION CODE : 03H：讀取資料。
06H：寫入資料。

DATA FIELD : 暫存器起始位址及欲讀取之 WORD 數或寫入之數值。

ERROR CHECK : 16bit CRC。

END OF FRAME : 至少 4 個字元的時間沒有傳送資料。

4.2 Bit Per Byte :

Start Bit	Data Bit	Parity	Stop	Frame
1	8	None	2	N,8,2
1	8	Even	1	E,8,1
1	8	Odd	1	O,8,1
1	8	None	1	N,8,1

5. 讀取暫存器命令：最大可讀取 8 個 Word。

Query :

Start of Frame	Address Field	Function Code	Start Address Hi	Start Address Lo	Number of Registers Hi	Number of Registers Lo	Error Check		End of Frame
	01~FFH	03H	nnH	nnH	nnH	nnH	CRC Lo	CRC Hi	
	1 Byte	1 Byte	2Byte		2 Byte		2 Byte		

Response :

Start of Frame	Address Field	Function Code	Number of Data Byte Count	D0、D1.. Dn (Hi,Lo,Hi,Lo....)	Error Check		End of Frame
	01~FFH	03H	Max:10H	Number of reg. * 2 Byte	CRC Lo	CRC Hi	
	1 Byte	1 Byte	1Byte	Max : 08H * 2 = 10H Byte	2 Byte		

** 須配合回傳資料格式 [CASE]選項 **

6. 寫入暫存器命令：為單一 WORD 寫入命令。

Query :

Start of Frame	Address Field	Function Code	Start Address Hi	Start Address Lo	Value Hi	Value Lo	Error Check		End of Frame
	01H~FFH	06H	0H	0 ~ 0AH	Setting Value		CRC Lo	CRC Hi	
	1 Byte	1 Byte	2Byte		2 Byte		2 Byte		

Response :

Start of Frame	Address Field	Function Code	Start Address Hi	Start Address Lo	Value Hi	Value Lo	Error Check		End of Frame
	01H~FFH	06H	0H	0 ~ 0AH	Setting Value		CRC Lo	CRC Hi	
	1 Byte	1 Byte	2Byte		2 Byte		2 Byte		

7. ERROR MESSAGE :

Start of Frame	Address Field	Function Code	Error Code	Error Check	End of Frame
	01H~FFH	83H or 86H		CRC Lo CRC Hi	
	1 Byte	1 Byte	1 Byte	2 Byte	

7.1 Function Code : 回應接收之 Function Code 但 MSB 設為 1。

- 7.2 Error Code : 01 : Error Function。
 02 : Error Data Address。
 03 : Error Data Value。

8. EXAMPLE :

- 8.1 讀取 Display Hi , Display Lo 資料：
 若 Display Hi : 9999 , Display Lo : 0。

Query :

Field Name	Example (Hex)
Meter Address	01
Function Code	03
Starting Address Hi	00
Starting Address Lo	03
Number of Word Hi	00
Number of Word Lo	02
CRC Lo	34
CRC Hi	0B

Response :

Field Name	Example (Hex)
Meter Address	01
Function Code	03
Data Byte Count	04
Data Hi (Address 3)	27
Data Lo	0f
Data Hi (Address 4)	00
Data Lo	00
CRC Lo	C0
CRC Hi	84

9. CRC 計算方式：

CRC 欄位為 2 個 16 進制 (Hex) Byte，從 ADDRESS FIELD 計算至 DATA FIELD 結束，若接收端計算之 CRC 與接收的不符，則表示資料錯誤。
從 ADDRESS FIELD 至 DATA FIELD 以 Message 表示。

9.1 計算方式：

1. 將 CRC 暫存器填入 FFFF(Hex)。
2. 將 CRC 暫存器低 8 位元與 Message 的第一個 Byte 做互斥或(Exclusive OR)，結果存入 CRC 暫存器。
3. 將 CRC 暫存器右移一個位元，CRC 暫存器最高位元填入 0，比較移出的位元(SLSB)。
4. 若 SLSB=0，重覆步驟 3。若 SLSB=1，將 CRC 暫存器與常數 A001(Hex)做互斥或，結果存入 CRC 暫存器。
5. 重覆步驟 3 及步驟 4，直到 8 位元都做完。
6. 重覆步驟 2~5，直到所有 Byte2 都做完。
7. 計算後之 CRC 暫存器之值，使用時需高低位元組互換填入 Message 之後。

9.2 查表方式：

回傳之 CRC 暫存器為 unsigned short int。
需傳入資料起始位址和資料長度，而回傳之 CRC 高低位元組已互換。

```
/*CRC Generation Function with 'C' language*/
unsigned short CRC16(ptMsg,usDatalen)
unsigned char *ptMsg;          /*message to calculate CRC upon*/
unsigned usDatalen;           /*number of bytes in message*/
{
    unsigned char uchCRCHi=0xFF; /*CRC high byte*/
    unsigned char uchCRCLo=0xFF; /*CRC low byte*/
    unsigned uIndex;
    while(usDatalen-->0) /*pass through message buffer*/
    {
        uIndex=uchCRCHi^*ptMsg++; /*calculate the CRC*/
        uchCRCHi=uchCRCLo^auchCRCHi[uIndex];
        uchCRCLo=auchCRCLo[uIndex];
    }
}
return (uchCRCHi<<8|uchCRCLo);
```

```
static unsigned char auchCRCHi[]={
0x00,0xc1,0x81,0x40,0x01,0xc0,0x80,0x41,0x01,0xc0,
0x80,0x41,0x00,0xc1,0x81,0x40,0x01,0xc0,0x80,0x41,
0x00,0xc1,0x81,0x40,0x00,0xc1,0x81,0x40,0x01,0xc0,
0x80,0x41,0x01,0xc0,0x80,0x41,0x00,0xc1,0x81,0x40,
0x00,0xc1,0x81,0x40,0x01,0xc0,0x80,0x41,0x00,0xc1,
0x81,0x40,0x01,0xc0,0x80,0x41,0x01,0xc0,0x80,0x41,
0x00,0xc1,0x81,0x40,0x01,0xc0,0x80,0x41,0x00,0xc1,
0x81,0x40,0x00,0xc1,0x81,0x40,0x01,0xc0,0x80,0x41,
0x00,0xc1,0x81,0x40,0x01,0xc0,0x80,0x41,0x01,0xc0,
0x80,0x41,0x00,0xc1,0x81,0x40,0x00,0xc1,0x81,0x40,
0x01,0xc0,0x80,0x41,0x01,0xc0,0x80,0x41,0x00,0xc1,
0x81,0x40,0x01,0xc0,0x80,0x41,0x00,0xc1,0x81,0x40,
0x00,0xc1,0x81,0x40,0x01,0xc0,0x80,0x41,0x01,0xc0,
0x80,0x41,0x00,0xc1,0x81,0x40,0x01,0xc0,0x80,0x41,
0x00,0xc1,0x81,0x40,0x00,0xc1,0x81,0x40,0x01,0xc0,
0x80,0x41,0x00,0xc1,0x81,0x40,0x01,0xc0,0x80,0x41,
0x01,0xc0,0x80,0x41,0x00,0xc1,0x81,0x40,0x01,0xc0,
0x80,0x41,0x00,0xc1,0x81,0x40,0x00,0xc1,0x81,0x40,
0x01,0xc0,0x80,0x41,0x01,0xc0,0x80,0x41,0x00,0xc1,
0x81,0x40,0x00,0xc1,0x81,0x40,0x01,0xc0,0x80,0x41,
0x00,0xc1,0x81,0x40,0x01,0xc0,0x80,0x41,0x01,0xc0,
0x80,0x41,0x00,0xc1,0x81,0x40};
```

```
static unsigned char auchCRCLo[]={
    0x00,0xc0,0xc1,0x01,0xc3,0x03,0x02,0xc2,0xc6,0x06,
    0x07,0xc7,0x05,0xc5,0xc4,0x04,0xcc,0x0c,0x0d,0xcd,
    0x0f,0xcf,0xce,0x0e,0x0a,0xca,0xcb,0x0b,0xc9,0x09,
    0x08,0xc8,0xd8,0x18,0x19,0xd9,0x1b,0xdb,0xda,0x1a,
    0x1e,0xde,0xdf,0x1f,0xdd,0x1d,0x1c,0xdc,0x14,0xd4,
    0xd5,0x15,0xd7,0x17,0x16,0xd6,0xd2,0x12,0x13,0xd3,
    0x11,0xd1,0xd0,0x10,0xf0,0x30,0x31,0xf1,0x33,0xf3,
    0xf2,0x32,0x36,0xf6,0xf7,0x37,0xf5,0x35,0x34,0xf4,
    0x3c,0xfc,0xfd,0x3d,0xff,0x3f,0x3e,0xfe,0xfa,0x3a,
    0x3b,0xfb,0x39,0xf9,0xf8,0x38,0x28,0xe8,0xe9,0x29,
    0xeb,0x2b,0x2a,0xea,0xee,0x2e,0x2f,0xef,0x2d,0xed,
    0xec,0x2c,0xe4,0x24,0x25,0xe5,0x27,0xe7,0xe6,0x26,
    0x22,0xe2,0xe3,0x23,0xe1,0x21,0x20,0xe0,0xa0,0x60,
    0x61,0xa1,0x63,0xa3,0xa2,0x62,0x66,0xa6,0xa7,0x67,
    0xa5,0x65,0x64,0xa4,0x6c,0xac,0xad,0x6d,0xaf,0x6f,
    0x6e,0xae,0xaa,0x6a,0x6b,0xab,0x69,0xa9,0xa8,0x68,
    0x78,0xb8,0xb9,0x79,0xbb,0x7b,0x7a,0xba,0xbe,0x7e,
    0x7f,0xbf,0x7d,0xbd,0xbc,0x7c,0xb4,0x74,0x75,0xb5,
    0x77,0xb7,0xb6,0x76,0x72,0xb2,0xb3,0x73,0xb1,0x71,
    0x70,0xb0,0x50,0x90,0x91,0x51,0x93,0x53,0x52,0x92,
    0x96,0x56,0x57,0x97,0x55,0x95,0x94,0x54,0x9c,0x5c,
    0x5d,0x9d,0x5f,0x9f,0x9e,0x5e,0x5a,0x9a,0x9b,0x5b,
    0x99,0x59,0x58,0x98,0x88,0x48,0x49,0x89,0x4b,0x8b,
    0x8a,0x4a,0x4e,0x8e,0x8f,0x4f,0x8d,0x4d,0x4c,0x8c,
    0x44,0x84,0x85,0x45,0x87,0x47,0x46,0x86,0x82,0x42,
    0x43,0x83,0x41,0x81,0x80,0x40};
```

10. 暫存器位址：設定值和顯示值皆不回傳小數點位置。

Function：讀取 (R)：03H ，寫入 (W)：06H。

Address	Wrod	Item(Description)	Function	Range(DEC)	Sign
0000H	1	Reserve	03		
0001H	1	Reserve	03		
0002H	1	Reserve	03		
0003H	1	dP Hi (Programmable Display Hi)	03	0~±9999	Signed
0004H	1	dP Lo (Programmable Display Lo)	03	0~±9999	Signed
0005H	1	Reserve	03		
0006H	1	Reserve	03		
0007H	1	(Display Value)	03	0~±10000	Signed
OFL：20000、-OFL：-20000					
0008H	1	dot (Dot Position)	03 or 06	0~3	Unsigned
0：0000.、1：000.0、2：00.00、3：0.000					
0009H	1	Reserve	03		
000AH	1	Reserve	03		
000BH	1	Reserve	03		
000CH	1	Reserve	03		
000DH	1	under (Under Input Lo)	03 or 06	0 or 1	Unsigned
000EH	1	Avg. (Average Display)	03 or 06	1~59	Unsigned
000FH	1	Baud(Baud Rate)	03 or 06	0~5	Unsigned
0：1200、1：2400、2：4800、3：9600、4：19200、5：38400					
0010H	1	Addr (Meter Address)	03 or 06	1~255	Unsigned
0011H	1	FrAE(Frame)	03 or 06	0~3	Unsigned
0：N,8,2、1：O,8,1、2：E,8,1、3：N,8,1					
0012H	1	Reserve	03		
0013H	1	Reserve	03		
0014H	1	Reserve	03		
0015H	1	Reserve	03		
0016H	1	RS485 Case	03 or 06	0~1	Unsigned
Two Word 0：Lo-Hi、1：Hi-Lo					
0017H	1	Reserve	03		
0018H	1	Display Adjustment %	03 or 06	799~1199	Unsigned
Unit：0.01%					

10.1 資料位址：浮點數格式。

Address	Wrod	Item(Description)	Function	Range(DEC)	Sign
1000H	2	(Display Value)	03	0~±10000	
1002H	2	Reserve	03		

10.2 資料位址：整數格式。

Address	Wrod	Item(Description)	Function	Range(DEC)	Sign
200H	2	(Display Value)	03	0~±10000	Signed
201H	2	dot (Dot Position)	03	0~4	Unsigned
0 : 0000. 、 1 : 000.0、 2 : 00.00、 3 : 0.000					
202H	2	Reserve	03		

11. 例題說明：

例：使用 Float:

Display = 9999讀取。

Query : WH

Field Name	(Hex)
Meter Address	01
Function Code	03
Starting Address Hi	10
Starting Address Lo	00
Number of Word Hi	00
Number of Word Lo	02
CRC Lo	C0
CRC Hi	CB

Response :

Case 0

Case 1

Field Name	(Hex)	(Hex)
Meter Address	01	01
Function Code	03	03
Data Byte Count	04	04
Data Hi (Address 0)	1C	9F
Data Lo	3C	46
Data Hi (Address 1)	9F	1C
Data Lo	46	3C
CRC Lo	D5	3D
CRC Hi	AD	23

*Float為直接之數值，不需考慮小數點和單位。

12. 浮點數說明：

IEEE 規格：4 BYTE = 32 BIT

1BIT	8 BIT	23 BIT
數值正(0)或負(1)	8BIT -127=EXP(2 ^ n)	尾數 23BIT

若使用 visual basic 語法如下：

先製訂使用者自訂的型態，因 Lset 只能用於自訂型態，如下：

```
Type usrflt
    Fvlu as Single
End Type
Type usrbyte
    Char(3) as byte
End Type
```

使用時先定義變數如：

```
Dim arya as usrbyte
Dim fla as usrflt
```

如 RS485 讀取的 byte 為 byte3->byte2->byte1->byte0：[注意 Case 設定]

程式如下：

```
arya.char(3)=Ascb(byte3)
arya.char(2)=Ascb(byte2)
arya.char(1)=Ascb(byte1)
arya.char(0)=Ascb(byte0)
Lset fla=arya
```

如此則 fla.fvlu 即為浮點數數值。

若使用 C 語言可用 union，或參照個別軟體之說明。