1 串列通訊協定

通訊協定採用標準的UART協定，串列傳輸速率是9600

2 通訊格式

2.1主機發送OneWire資料群組：

OneWire資料會分成兩個群組:

1. System Information Group
2. EEPROM Information Group

2.2 主機發送OneWire資料格式

//========One Wire Transmiting Data Structure=======================================================

//typedef struct{

// static uint cStart1 = 0x80f8; //前導碼 // send High byte first, then send Low byte second.

// static uint cStart2 = 0x80A0; //資料群組 // send High byte first, then send Low byte second.

// // 0x80A0 : system information 資料,

// // 0x80D0 : EEPROM information,

// unsigned uint LenExpected; //數據長度 ; send High byte first, then send Low byte second.

// unsigned char DataBuf[LenExpected];//數據內容

// unsigned char LRCDataHigh; //ModBus CRC16 High byte, calculating only include LenExpected and DataBuf

// unsigned char LRCDataLow; //ModBus CRC16 Low byte, calculating only include LenExpected and DataBuf

// static uint cEND1= 0x70f7; //結束字組 1 // send High byte first, then send Low byte second.

// static uint cEND1= 0x70f7; //結束字組 2 // send High byte first, then send Low byte second.

//} OneWireProtocolPacket;

* 1. **System Information Data Examples**

80 F8 80 A0 00 46 00 03 00 02 10 00 21 00 00 E3 00 05 00 88 00 00 00 00 07 C6 06 54 06 C2 0D 15 00 00 00 00 00 00 00 00 00 00 00 01 07 C8 07 C0 00 00 00 00 06 58 06 45 06 C6 06 B8 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 **53 A8** 70 F7 70 F7

1. 綠底字為 前導碼及結束碼
2. 紫紅底字為 數據長度， 0x0046 = 70 bytes 資料
3. 白底底線為 數據。
4. 紅字 0x53A8 為 Modbus CRC16, 即包含數據長度及數據，所計算出來的。
   1. **EEPROM Information Data Examples**



1. 綠底字為 前導碼及結束碼
2. EEPROM 無數據長度
3. 白底底線為 數據。
4. 紅字 0x30AF 為 Modbus CRC16, 即數據所計算出來的。
   1. Modbus CRC 16 計算function

///////////////////////////////////////////////////////////////////

//ModBus Crc16 Table

private static ushort[] CrcTable = {

0X0000, 0XC0C1, 0XC181, 0X0140, 0XC301, 0X03C0, 0X0280, 0XC241,

0XC601, 0X06C0, 0X0780, 0XC741, 0X0500, 0XC5C1, 0XC481, 0X0440,

0XCC01, 0X0CC0, 0X0D80, 0XCD41, 0X0F00, 0XCFC1, 0XCE81, 0X0E40,

0X0A00, 0XCAC1, 0XCB81, 0X0B40, 0XC901, 0X09C0, 0X0880, 0XC841,

0XD801, 0X18C0, 0X1980, 0XD941, 0X1B00, 0XDBC1, 0XDA81, 0X1A40,

0X1E00, 0XDEC1, 0XDF81, 0X1F40, 0XDD01, 0X1DC0, 0X1C80, 0XDC41,

0X1400, 0XD4C1, 0XD581, 0X1540, 0XD701, 0X17C0, 0X1680, 0XD641,

0XD201, 0X12C0, 0X1380, 0XD341, 0X1100, 0XD1C1, 0XD081, 0X1040,

0XF001, 0X30C0, 0X3180, 0XF141, 0X3300, 0XF3C1, 0XF281, 0X3240,

0X3600, 0XF6C1, 0XF781, 0X3740, 0XF501, 0X35C0, 0X3480, 0XF441,

0X3C00, 0XFCC1, 0XFD81, 0X3D40, 0XFF01, 0X3FC0, 0X3E80, 0XFE41,

0XFA01, 0X3AC0, 0X3B80, 0XFB41, 0X3900, 0XF9C1, 0XF881, 0X3840,

0X2800, 0XE8C1, 0XE981, 0X2940, 0XEB01, 0X2BC0, 0X2A80, 0XEA41,

0XEE01, 0X2EC0, 0X2F80, 0XEF41, 0X2D00, 0XEDC1, 0XEC81, 0X2C40,

0XE401, 0X24C0, 0X2580, 0XE541, 0X2700, 0XE7C1, 0XE681, 0X2640,

0X2200, 0XE2C1, 0XE381, 0X2340, 0XE101, 0X21C0, 0X2080, 0XE041,

0XA001, 0X60C0, 0X6180, 0XA141, 0X6300, 0XA3C1, 0XA281, 0X6240,

0X6600, 0XA6C1, 0XA781, 0X6740, 0XA501, 0X65C0, 0X6480, 0XA441,

0X6C00, 0XACC1, 0XAD81, 0X6D40, 0XAF01, 0X6FC0, 0X6E80, 0XAE41,

0XAA01, 0X6AC0, 0X6B80, 0XAB41, 0X6900, 0XA9C1, 0XA881, 0X6840,

0X7800, 0XB8C1, 0XB981, 0X7940, 0XBB01, 0X7BC0, 0X7A80, 0XBA41,

0XBE01, 0X7EC0, 0X7F80, 0XBF41, 0X7D00, 0XBDC1, 0XBC81, 0X7C40,

0XB401, 0X74C0, 0X7580, 0XB541, 0X7700, 0XB7C1, 0XB681, 0X7640,

0X7200, 0XB2C1, 0XB381, 0X7340, 0XB101, 0X71C0, 0X7080, 0XB041,

0X5000, 0X90C1, 0X9181, 0X5140, 0X9301, 0X53C0, 0X5280, 0X9241,

0X9601, 0X56C0, 0X5780, 0X9741, 0X5500, 0X95C1, 0X9481, 0X5440,

0X9C01, 0X5CC0, 0X5D80, 0X9D41, 0X5F00, 0X9FC1, 0X9E81, 0X5E40,

0X5A00, 0X9AC1, 0X9B81, 0X5B40, 0X9901, 0X59C0, 0X5880, 0X9841,

0X8801, 0X48C0, 0X4980, 0X8941, 0X4B00, 0X8BC1, 0X8A81, 0X4A40,

0X4E00, 0X8EC1, 0X8F81, 0X4F40, 0X8D01, 0X4DC0, 0X4C80, 0X8C41,

0X4400, 0X84C1, 0X8581, 0X4540, 0X8701, 0X47C0, 0X4680, 0X8641,

0X8201, 0X42C0, 0X4380, 0X8341, 0X4100, 0X81C1, 0X8081, 0X4040 };

public static UInt16 ComputeModBusCrc16(byte[] data)

{

ushort crc = 0xFFFF;

foreach (byte datum in data)

{

crc = (ushort)((crc >> 8) ^ CrcTable[(crc ^ datum) & 0xFF]);

}

return crc;

}

1. **System Information Data 解析**

80 F8 80 A0 00 46 00 03 00 02 10 00 21 00 00 E3 00 05 00 88 00 00 00 00 07 C6 06 54 06 C2 0D 15 00 00 00 00 00 00 00 00 00 00 00 01 07 C8 07 C0 00 00 00 00 06 58 06 45 06 C6 06 B8 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 **53 A8** 70 F7 70 F7

去掉前導碼及數據長度，第一組 word為 0x0003.: 0x00, high byte, 0x03:low byte

word 01 : G\_SysModeStatusCode

word 02 : G\_Module\_Status

word 03 : G\_Module\_Function\_Status

word 04 : G\_Add\_Module\_Function\_Status

word 05 : G\_Device\_Interface\_Status

word 06 : G\_Extend\_Device\_Interface\_Status

word 07 : G\_Capacity\_Var\_Bits\_Status

word 08 : G\_DSG\_Current\_ADC

word 09 : G\_CHG\_Current\_ADC

word 10 : G\_VBAT\_ADC

word 11 : G\_NTC1\_ADC

word 12 : G\_NTC2\_ADC

word 13 : G\_Vref\_mVoltage

word 14 : G\_Current\_Capacity

word 15 : G\_Hold\_Capacity

word 16 : G\_AVG\_DSG\_Current\_ADC

word 17 : G\_RECORD\_DATA\_COUNT

word 18 : G\_MAX\_DSG\_C\_ADC\_RECORD

word 19 : G\_MAX\_CHG\_C\_ADC\_RECORD

word 20 : G\_MAX\_VBAT\_ADC\_RECORD

word 21 : G\_MIN\_VBAT\_ADC\_RECORD

word 22 : G\_MAX\_VBAT\_SocLo\_ADC\_RECORD

word 23 : G\_MAX\_VBAT\_SocHi\_ADC\_RECORD

word 24 : G\_DSG\_MAX\_TH1\_ADC\_RECORD

word 25 : G\_DSG\_MIN\_TH1\_ADC\_RECORD

word 26 : G\_DSG\_MAX\_TH2\_ADC\_RECORD

word 27 : G\_DSG\_MIN\_TH2\_ADC\_RECORD

word 28 : G\_CHG\_Cycle\_Count\_RECORD

word 29 : G\_CHG\_ADC\_AccumulatingQ\_RECORD\_Hi

word 30 : G\_CHG\_ADC\_AccumulatingQ\_RECORD\_Lo

word 31 : G\_DSG\_ADC\_AccumulatingQ\_RECORD\_Hi

word 32 : G\_DSG\_ADC\_AccumulatingQ\_RECORD\_Lo

word 33 : G\_DSG\_Cycle\_Count\_RECORD

word 34 : G\_CHG\_MAX\_TH1\_ADC\_RECORD

word 35 : G\_CHG\_MIN\_TH1\_ADC\_RECORD

word 36 : G\_CHG\_MAX\_TH2\_ADC\_RECORD

word 37 : G\_CHG\_MIN\_TH2\_ADC\_RECORD

word 38 : G\_OVP\_TIMES

word 39 : G\_UVP\_TIMES

word 40 : G\_COCP\_TIMES

word 41 : G\_DOCP\_TIMES

word 42 : G\_TH1\_UTP\_TIMES

word 43 : G\_TH2\_UTP\_TIMES

word 44 : G\_CHG\_TH1\_OTP\_TIMES

word 45 : G\_CHG\_TH2\_OTP\_TIMES

word 46 : G\_DSG\_TH1\_LOW\_CURRENT\_OTP\_TIMES

word 47 : G\_DSG\_TH1\_HIGH\_CURRENT\_OTP\_TIMES

word 48 : G\_DSG\_TH2\_LOW\_CURRENT\_OTP\_TIMES

word 49 : G\_DSG\_TH2\_HIGH\_CURRENT\_OTP\_TIMES

word 50 : G\_RECORD\_2nd\_DATA\_COUNT

word 51 : G\_REAL\_FCC\_UPDATE\_TIMES

word 52 : G\_Real\_FCC\_In\_CoulombCounter

word 53 : G\_Derating\_FCC\_In\_CoulombCounter

word 54 : G\_RM\_In\_CoulombCounter

word 55 : G\_Capacity\_In\_CoulombCounter

word 56 : G\_STATIC\_OVER\_VOLTAGE\_HOURS\_TIMES

word 57 : G\_CHG\_OVER\_VOLTAGE\_HOURS\_TIMES

word 58 : G\_OVER\_LOADING\_MINUTES\_TIMES

word 59 : G\_FASTER\_CHARGING\_MINUTES\_TIMES

word 60 : G\_CHARGING\_IN\_HIGH\_TEMP\_HOURS\_TIMES

word 61 : G\_CHARGING\_IN\_LOW\_TEMP\_HOURS\_TIMES

word 62 : G\_STORE\_IN\_LOW\_TEMP1\_HOURS\_TIMES

word 63 : G\_STORE\_IN\_LOW\_TEMP2\_HOURS\_TIMES

word 64 : G\_STORE\_IN\_LOW\_TEMP3\_HOURS\_TIMES

word 65 : G\_STORE\_IN\_LOW\_TEMP4\_HOURS\_TIMES

word 66 : G\_STORE\_IN\_HIGH\_TEMP1\_MINUTES\_TIMES

word 67 : G\_STORE\_IN\_HIGH\_TEMP2\_MINUTES\_TIMES

word 68 : G\_STORE\_IN\_HIGH\_TEMP3\_MINUTES\_TIMES

word 69 : G\_STORE\_IN\_HIGH\_TEMP4\_MINUTES\_TIMES

word 70 : G\_STORE\_IN\_HIGH\_TEMP5\_MINUTES\_TIMES

word 71 : G\_RECORD\_3rd\_TRACKING\_DATA\_COUNT

word 72 : G\_NTC1\_Real\_0p1\_KelvinDegree

word 73 : G\_NTC2\_Real\_0p1\_KelvinDegree

word 74 : G\_Last\_FCC\_Updated\_Values

word 75 : G\_AVG\_Battery\_Voltage\_ADC

1. **EEPROM Information Data 解析**



去掉前導碼，0x3E 為offset 0, 0xE4 為0ffset 1

// Data Offset

// from hi byte to low byte

// float:4bytes,

// char(signed char): 1 byte. uchart(unsigned char): 1bytes

// int(signed int):2bytes. unit(unsigned int): 2bytes

// long(signed long):4bytes. ulong(unsigned long):4bytes

CHG\_mA\_To\_ADC\_Factor\_Offset 0 // 0x0 ; Data Type: float

DSG\_mA\_To\_ADC\_Factor\_Offset 4 // 0x4 ; Data Type: float

VBAT\_mV\_To\_ADC\_Factor\_Offset 8 // 0x8 ; Data Type: float

Thermistor\_mV\_To\_ADC\_Factor\_Offset 12 // 0xC ; Data Type: float

DSG\_OP\_ADC\_OFFSET\_Offset 16 // 0x10 ; Data Type: char

CHG\_OP\_ADC\_OFFSET\_Offset 17 // 0x11 ; Data Type: char

VBAT\_ADC\_OFFSET\_Offset 18 // 0x12 ; Data Type: char

NTC1\_ADC\_OFFSET\_Offset 19 // 0x13 ; Data Type: char

NTC2\_ADC\_OFFSET\_Offset 20 // 0x14 ; Data Type: char

SOC\_ADC\_OFFSET\_Offset 21 // 0x15 ; Data Type: char

RESERVED\_CAL\_FOR\_OFFSET1\_Offset 22 // 0x16 ; Data Type: ulong

RESERVED\_CAL\_FOR\_OFFSET2\_Offset 26 // 0x1A ; Data Type: ulong

RESERVED\_CAL\_FOR\_OFFSET3\_Offset 30 // 0x1E ; Data Type: uint

MAX\_DSG\_C\_ADC\_RECORD\_EEPROM\_Offset 32 // 0x20 ; Data Type: uint

MAX\_CHG\_C\_ADC\_RECORD\_EEPROM\_Offset 34 // 0x22 ; Data Type: uint

MAX\_VBAT\_ADC\_RECORD\_EEPROM\_Offset 36 // 0x24 ; Data Type: uint

MIN\_VBAT\_ADC\_RECORD\_EEPROM\_Offset 38 // 0x26 ; Data Type: uint

MAX\_VBAT\_SocLo\_ADC\_RECORD\_EEPROM\_Offset 40 // 0x28 ; Data Type: uint

MAX\_VBAT\_SocHi\_ADC\_RECORD\_EEPROM\_Offset 42 // 0x2A ; Data Type: uint

DSG\_MAX\_TH1\_ADC\_RECORD\_EEPROM\_Offset 44 // 0x2C ; Data Type: uint

DSG\_MIN\_TH1\_ADC\_RECORD\_EEPROM\_Offset 46 // 0x2E ; Data Type: uint

DSG\_MAX\_TH2\_ADC\_RECORD\_EEPROM\_Offset 48 // 0x30 ; Data Type: uint

DSG\_MIN\_TH2\_ADC\_RECORD\_EEPROM\_Offset 50 // 0x32 ; Data Type: uint

RECORD\_DATA\_COUNT\_EEPROM\_Offset 52 // 0x34 ; Data Type: uint

CHG\_Cycle\_Count\_RECORD\_EEPROM\_Offset 54 // 0x36 ; Data Type: uint

CHG\_ADC\_AccumulatingQ\_RECORD\_EEPROM\_Offset 56 // 0x38 ; Data Type: ulong

DSG\_ADC\_AccumulatingQ\_RECORD\_EEPROM\_Offset 60 // 0x3C ; Data Type: ulong

DSG\_Cycle\_Count\_RECORD\_EEPROM\_Offset 64 // 0x40 ; Data Type: uint

RESERVED\_Recording\_FOR\_OFFSET1\_Offset 66 // 0x42 ; Data Type: uint

CHG\_MAX\_TH1\_ADC\_RECORD\_EEPROM\_Offset 68 // 0x44 ; Data Type: uint

CHG\_MIN\_TH1\_ADC\_RECORD\_EEPROM\_Offset 70 // 0x46 ; Data Type: uint

CHG\_MAX\_TH2\_ADC\_RECORD\_EEPROM\_Offset 72 // 0x48 ; Data Type: uint

CHG\_MIN\_TH2\_ADC\_RECORD\_EEPROM\_Offset 74 // 0x4A ; Data Type: uint

G\_OVP\_TIMES\_RECORD\_EEPROM\_Offset 76 // 0x4C ; Data Type: uint

G\_UVP\_TIMES\_RECORD\_EEPROM\_Offset 78 // 0x4E ; Data Type: uint

G\_COCP\_TIMES\_RECORD\_EEPROM\_Offset 80 // 0x50 ; Data Type: uint

G\_DOCP\_TIMES\_RECORD\_EEPROM\_Offset 82 // 0x52 ; Data Type: uint

G\_TH1\_UTP\_TIMES\_RECORD\_EEPROM\_Offset 84 // 0x54 ; Data Type: uint

G\_TH2\_UTP\_TIMES\_RECORD\_EEPROM\_Offset 86 // 0x56 ; Data Type: uint

G\_CHG\_TH1\_OTP\_TIMES\_RECORD\_EEPROM\_Offset 88 // 0x58 ; Data Type: uint

G\_CHG\_TH2\_OTP\_TIMES\_RECORD\_EEPROM\_Offset 90 // 0x5A ; Data Type: uint

RECORD\_2nd\_DATA\_COUNT\_EEPROM\_Offset 92 // 0x5C ; Data Type: uint

RESERVED\_Recording\_FOR\_OFFSET3\_Offset 94 // 0x5E ; Data Type: uint

SystemSetting\_Info\_Byte1\_Offset 96 // 0x60 ; Data Type: uchar

SystemSetting\_Info\_Byte2\_Offset 97 // 0x61 ; Data Type: uchar

System\_Control\_Bit\_EEPROM\_Offset 98 // 0x62 ; Data Type: uchar

SystemSetting\_Info\_Byte3\_Offset 99 // 0x63 ; Data Type: uchar

VERSION\_Offset 100 // 0x64 ; Data Type: uchar

MINOR\_VERSION\_Offset 101 // 0x65 ; Data Type: uchar

EEPROM\_VERSION\_Offset 102 // 0x66 ; Data Type: uchar

RESERVED\_VERSION\_Offset 103 // 0x67 ; Data Type: uchar

NUMBER\_OF\_PARALLEL\_CELLS\_Offset 104 // 0x68 ; Data Type: uchar

NUMBER\_OF\_SERIES\_CELLS\_Offset 105 // 0x69 ; Data Type: uchar

MANUFACTURE\_DATE\_Offset 106 // 0x6A ; Data Type: uint

SERIAL\_NUMBER\_Offset 108 // 0x6C ; Data Type: uint

CELL\_TYPE\_LENGTH\_Offset 110 // 0x6E ; Data Type: uchar

CELL\_TYPE\_Offset 111 // 0x6F ; Data Type: uchar

MANUFACTURE\_NAME\_LENGTH\_Offset 119 // 0x77 ; Data Type: uchar

MANUFACTURE\_NAME\_Offset 120 // 0x78 ; Data Type: uchar

ADC\_CURRENT\_DETECT\_FOR\_DSG\_STATUS\_Offset 128 // 0x80 ; Data Type: uint

ADC\_CURRENT\_DETECT\_FOR\_CHG\_STATUS\_Offset 130 // 0x82 ; Data Type: uint

ADC\_DOC\_PROTECTION\_Offset 132 // 0x84 ; Data Type: uint

ADC\_COC\_PROTECTION\_Offset 134 // 0x86 ; Data Type: uint

ADC\_DSG\_HIGH\_CURRENT\_DETECT\_Offset 136 // 0x88 ; Data Type: uint

ADC\_CHG\_CHARGER\_TAPE\_CURRENT\_Offset 138 // 0x8A ; Data Type: uint

ADC\_AccQ\_FOR\_CHG\_TH\_Offset 140 // 0x8C ; Data Type: ulong

Cycle\_Count\_FOR\_CHG\_1st\_TH\_Offset 144 // 0x90 ; Data Type: uint

Cycle\_Count\_FOR\_CHG\_2nd\_TH\_Offset 146 // 0x92 ; Data Type: uint

ADC\_BATTERY\_OV\_PROTECTION\_Offset 148 // 0x94 ; Data Type: uint

ADC\_BATTERY\_OV\_RELEASE\_Offset 150 // 0x96 ; Data Type: uint

ADC\_BATTERY\_UV\_PROTECTION\_Offset 152 // 0x98 ; Data Type: uint

ADC\_BATTERY\_UV\_RELEASE\_Offset 154 // 0x9A ; Data Type: uint

ADC\_DSG\_OT\_HIGH\_PROTECTION\_Offset 156 // 0x9C ; Data Type: uint

ADC\_DSG\_OT\_HIGH\_RELEASE\_Offset 158 // 0x9E ; Data Type: uint

ADC\_DSG\_OT\_LOW\_PROTECTION\_Offset 160 // 0xA0 ; Data Type: uint

ADC\_DSG\_OT\_LOW\_RELEASE\_Offset 162 // 0xA2 ; Data Type: uint

ADC\_CHG\_OT\_PROTECTION\_Offset 164 // 0xA4 ; Data Type: uint

ADC\_CHG\_OT\_RELEASE\_Offset 166 // 0xA6 ; Data Type: uint

ADC\_UT\_PROTECTION\_Offset 168 // 0xA8 ; Data Type: uint

ADC\_UT\_RELEASE\_Offset 170 // 0xAA ; Data Type: uint

ADC\_INITIAL\_CHARGING\_TEMP\_RANGE\_HI\_Offset 172 // 0xAC ; Data Type: uint

ADC\_INITIAL\_CHARGING\_TEMP\_RANGE\_LO\_Offset 174 // 0xAE ; Data Type: uint

ADC\_LOW\_TEMP\_SOC\_CHARGING\_RANGE\_HI\_Offset 176 // 0xB0 ; Data Type: uint

ADC\_LOW\_TEMP\_SOC\_CHARGING\_RANGE\_LO\_Offset 178 // 0xB2 ; Data Type: uint

ADC\_AccQ\_FOR\_DSG\_TH\_Offset 180 // 0xB4 ; Data Type: ulong

DSG\_CAPACITY\_DISPLAY\_TH\_1\_Offset 184 // 0xB8 ; Data Type: uint

DSG\_CAPACITY\_DISPLAY\_TH\_2\_Offset 186 // 0xBA ; Data Type: uint

DSG\_CAPACITY\_DISPLAY\_TH\_3\_Offset 188 // 0xBC ; Data Type: uint

DSG\_CAPACITY\_DISPLAY\_TH\_4\_Offset 190 // 0xBE ; Data Type: uint

DSG\_CAPACITY\_DISPLAY\_TH\_5\_Offset 192 // 0xC0 ; Data Type: uint

CHG\_CAPACITY\_DISPLAY\_TH\_1\_Offset 194 // 0xC2 ; Data Type: uint

CHG\_CAPACITY\_DISPLAY\_TH\_2\_Offset 196 // 0xC4 ; Data Type: uint

CHG\_CAPACITY\_DISPLAY\_TH\_3\_Offset 198 // 0xC6 ; Data Type: uint

CHG\_CAPACITY\_DISPLAY\_TH\_4\_Offset 200 // 0xC8 ; Data Type: uint

\_ADC\_LOOKUP\_DSG\_TEMP\_1\_TH\_\_Offset 202 // 0xCA ; Data Type: uint

\_ADC\_LOOKUP\_DSG\_TEMP\_2\_TH\_\_Offset 204 // 0xCC ; Data Type: uint

\_ADC\_LOOKUP\_DSG\_TEMP\_3\_TH\_\_Offset 206 // 0xCE ; Data Type: uint

\_ADC\_LOOKUP\_CHG\_TEMP\_1\_TH\_\_Offset 208 // 0xD0 ; Data Type: uint

\_ADC\_LOOKUP\_CHG\_TEMP\_2\_TH\_\_Offset 210 // 0xD2 ; Data Type: uint

\_ADC\_DSG\_CURRENT\_LOOKUP\_OCV\_TABLE\_TH1\_\_Offset 212 // 0xD4 ; Data Type: uint

\_ADC\_DSG\_CURRENT\_LOOKUP\_OCV\_TABLE\_TH2\_\_Offset 214 // 0xD6 ; Data Type: uint

\_ADC\_DSG\_CURRENT\_LOOKUP\_OCV\_TABLE\_TH3\_\_Offset 216 // 0xD8 ; Data Type: uint

\_ADC\_DSG\_CURRENT\_LOOKUP\_OCV\_TABLE\_TH4\_\_Offset 218 // 0xDA ; Data Type: uint

\_ADC\_DSG\_CURRENT\_LOOKUP\_OCV\_TABLE\_TH5\_\_Offset 220 // 0xDC ; Data Type: uint

\_ADC\_DSG\_CURRENT\_LOOKUP\_OCV\_TABLE\_TH6\_\_Offset 222 // 0xDE ; Data Type: uint

AUTO\_ENTRY\_SLEEPING\_DELAY\_MINUTES\_Offset 224 // 0xE0 ; Data Type: uint

CAPACITY\_DIFFERENCE\_VALUES\_Offset 226 // 0xE2 ; Data Type: uint

PROJECT\_NAME\_MAX\_LENGTH\_Offset 228 // 0xE4 ; Data Type: uchar

PROJECT\_NAME\_Offset 229 // 0xE5 ; Data Type: uchar

MCU\_UID\_MAX\_LENGTH\_Offset 253 // 0xFD ; Data Type: uchar

MCU\_UID\_Offset 254 // 0xFE ; Data Type: uchar

BAR\_CODE\_REAL\_STORE\_LENGTH\_Offset 266 // 0x10A; Data Type: uchar

BAR\_CODE\_MAX\_LENGTH\_Offset 267 // 0x10B; Data Type: uchar

BAR\_CODE\_Offset 268 // 0x10C; Data Type: uchar

DESIGN\_CAPACITY\_mAH\_Offset 292 // 0x124; Data Type: uint

Last\_Record\_FCC\_mAh\_RECORD\_EEPROM\_Offset 294 // 0x126; Data Type: uint

REAL\_FCC\_UPDATE\_TIMES\_RECORD\_EEPROM\_Offset 296 // 0x128; Data Type: uint

REAL\_FCC\_mAH\_RECORD\_EEPROM\_Offset 298 // 0x12A; Data Type: uint

STATIC\_OVER\_VOLTAGE\_HOURS\_TIMES\_RECORD\_EEPROM\_Offset 300 // 0x12C; Data Type: uint

CHG\_OVER\_VOLTAGE\_HOURS\_TIMES\_RECORD\_EEPROM\_Offset 302 // 0x12E; Data Type: uint

OVER\_LOADING\_MINUTES\_TIMES\_RECORD\_EEPROM\_Offset 304 // 0x130; Data Type: uint

FASTER\_CHARGING\_MINUTES\_TIMES\_RECORD\_EEPROM\_Offset 306 // 0x132; Data Type: uint

CHARGING\_IN\_HIGH\_TEMP\_HOURS\_TIMES\_RECORD\_EEPROM\_Offset 308 // 0x134; Data Type: uint

CHARGING\_IN\_LOW\_TEMP\_HOURS\_TIMES\_RECORD\_EEPROM\_Offset 310 // 0x136; Data Type: uint

STORE\_IN\_LOW\_TEMP1\_HOURS\_TIMES\_RECORD\_EEPROM\_Offset 312 // 0x138; Data Type: uint

STORE\_IN\_LOW\_TEMP2\_HOURS\_TIMES\_RECORD\_EEPROM\_Offset 314 // 0x13A; Data Type: uint

STORE\_IN\_LOW\_TEMP3\_HOURS\_TIMES\_RECORD\_EEPROM\_Offset 316 // 0x13C; Data Type: uint

STORE\_IN\_LOW\_TEMP4\_HOURS\_TIMES\_RECORD\_EEPROM\_Offset 318 // 0x13E; Data Type: uint

G\_DSG\_TH1\_LOW\_CURRENT\_OTP\_TIMES\_RECORD\_EEPROM\_Offset 320 // 0x140; Data Type: uint

G\_DSG\_TH1\_HIGH\_CURRENT\_OTP\_TIMES\_RECORD\_EEPROM\_Offset 322 // 0x142; Data Type: uint

G\_DSG\_TH2\_LOW\_CURRENT\_OTP\_TIMES\_RECORD\_EEPROM\_Offset 324 // 0x144; Data Type: uint

G\_DSG\_TH2\_HIGH\_CURRENT\_OTP\_TIMES\_RECORD\_EEPROM\_Offset 326 // 0x146; Data Type: uint

STORE\_IN\_HIGH\_TEMP1\_MINUTES\_TIMES\_RECORD\_EEPROM\_Offset 328 // 0x148; Data Type: uint

STORE\_IN\_HIGH\_TEMP2\_MINUTES\_TIMES\_RECORD\_EEPROM\_Offset 330 // 0x14A; Data Type: uint

STORE\_IN\_HIGH\_TEMP3\_MINUTES\_TIMES\_RECORD\_EEPROM\_Offset 332 // 0x14C; Data Type: uint

STORE\_IN\_HIGH\_TEMP4\_MINUTES\_TIMES\_RECORD\_EEPROM\_Offset 334 // 0x14E; Data Type: uint

STORE\_IN\_HIGH\_TEMP5\_MINUTES\_TIMES\_RECORD\_EEPROM\_Offset 336 // 0x150; Data Type: uint

RECORD\_3rd\_TRACKING\_DATA\_COUNT\_EEPROM\_Offset 338 // 0x152; Data Type: uint

EEPROM\_END\_DATA\_POSITION\_Offset 340 // 0x154; Data Type: uchar