

# COMMUNICATION WITH THE SuperBrain & DigiPoint

Last Update 12.01.2010

## MODBUS Protocol

The **SuperBrain & DigiPoint** has a serial interface port allowing direct interface with an external communication network supporting the MODBUS Protocol.

MODBUS is an Industry Standard, widely known and commonly used communications protocol. Using MODBUS provides communication between a PC and up to 247 Powermeter slaves on a common line- the PC being the **master** and the powermeters the **slaves**. The PC initiates the transaction (either a query or broadcast) and the Powermeter/s responds. Powermeters respond to the **master** PC's request, but will not initiate any transmission on its own. The PC sends a single Query transaction and the Powermeter responds in a single response frame and is capable of only one query and one response at a time

### 1.1 — MODBUS Framing

#### 1.1.1— RTU Transmission Mode

MODBUS uses the standard Remote Terminal Unit (RTU) transmission mode. RTU mode sends data in 8-bit binary EVEN parity or 8-bit binary NO parity data format. For the **SuperBrain & DigiPoint** to successfully communicate, choose one in the communication Set Up.

Field	No. of bits
Start bit	1
Data bits	8
Parity	1
Stop bit	1

Table 1-1 RTU Data Format

## 1.1.2 — The RTU Frame Format

Query and response information is sent in frames. Each frame contains:

Address

Function (See Section 1.1.4 for descriptions of functions),

Data

Check.

Address	Function	Data	Check
8 bits	8 bits	N * 8 bits	16 bits

Table 1-2 R T U Message Frame Format

If the receiving device (Powermeter) detects a time laps of five characters, then it will assume the message is incomplete and will flush the frame. The device then assumes that the next byte received will be an address. The maximum query and response message length is 256 bytes including check characters.

## 1.1.3 — Address Field

Each Powermeter is designated in a network system by a user assigned address. The Address can be any number between 1 and 255. The Powermeter will only respond to it's own specifacally assigned address.

## 1.1.4 — Function Field

The function field contains the code that tells the Powermeter what action to perform.

The ***SuperBrain & DigiPoint*** uses and responds to four standard Message Format Functions.

**Function 03**

**Function 04**

**Function 06**

**Function 16**

Function	Meaning in MODBUS	Action
Function 03	Read holding register	Obtain data from Powermeter (Read register)
Function 04	Read input register	Obtain data from Powermeter (Read register)
Function 06	Preset single register	Transmit data to Powermeter (Write single register)
Function 16	Preset multiple register	Transmit data to Powermeter (Write multiple register)

Table 1-3 **Function Codes**

### 1.1.5 — Data Field

The Data field contains the body of the message and contains instructions from the PC **master** to the Powermeter **slave** to perform a particular action or respond to a query. The reply message from the Powermeter will be information contained in one or more of its registers.

### 1.1.6 — Check Field

The error check field contains the result of Cyclical Redundancy Check (CRC). The start of the message is ignored in calculating the CRC.

For more detailed information on CRC, refer to the MODBUS Protocol Reference Guide.

## 1.2 — Registers for *SuperBrain & DigiPoint*

The *SuperBrain & DigiPoint* is capable of supporting either Function 03 or Function 04 Message Format(See Table 1-3). In a reply to a query from the PC **master** for a reading from a particular field, the response from the Powermeter can be either in Format 03 or Format 04 but will depend on which Format the query was originally sent.

The difference is significant because by using Function 03 the SuperBrain & DigiPoint will only send the INTERGER part of the field value requested and the PC **master** will only display the INTERGER part of the field value.

Function 04 on the other hand, is capable of sending two separate halves of the full FLOAT requested information (each half contained in a separate register). Then it is the task of the PC **master** to merge the two halves into a full FLOAT reply. (For more detailed information See IEEE Standard 754 Floating-Point).

<b>MODBUS Register</b>	<b>Field Description</b>	<b>Type</b>	<b>ITEM No (Uniart)</b>
1-2	Parameter #1 (File 0)	Read/Write	1
3-4	Parameter #2 (file 0)	Read/Write	2
↓	↓	↓	↓
255-256	Parameter #128 (File 0)	Read/Write	128
257-258	Parameter #1 (File 1)	Read/Write	129
↓	↓	↓	↓
511-512	Parameter #128 (File 1)	Read/Write	256
513-514	Parameter #1 (File 2)	Read/Write	257
↓	↓	↓	↓
767-768	Parameter #128 (File 2)	Read/Write	384
769-770	Parameter #1 (File 3)	Read/Write	385
↓	↓	↓	↓
1023-1024	Parameter #128 (File 3)	Read/Write	512
1025-1026	Parameter #1 (File 4)	Read/Write	513
↓	↓	↓	↓
1279-1280	Parameter #128 (File 4)	Read/Write	640
1281-1282	Parameter #1 (File 5)	Read/Write	641
↓	↓	↓	↓
1535-1536	Parameter #128 (File 5)	Read/Write	768
1537-1538	Parameter #1 (File 6)	Read/Write	769
↓	↓	↓	↓
1791-1792	Parameter #128 (File 6)	Read/Write	896
1793-1794	Parameter #1 (File 7)	Read/Write	897
↓	↓	↓	↓
2047-2048	Parameter #128 (File 7)	Read/Write	1024
2049-2050	Parameter #1 (File 8)	Read/Write	1025
↓	↓	↓	↓
2303-2304	Parameter #128 (File 8)	Read/Write	1152
4097-4098	Parameter #1 (File 16)	Read/Write	2049
↓	↓	↓	↓
4351-4352	Parameter #128 (File 16)	Read/Write	2176
6145-6146	Parameter #1 (File 24)	Read/Write	3073
↓	↓	↓	↓
6399-6400	Parameter #128 (File 24)	Read/Write	3200

<b>MODBUS Register</b>	<b>Field Description</b>	<b>Type</b>	<b>ITEM No (Uniart)</b>
8193-8194	Parameter #1 (File 32)	Read/Write	4097
↓	↓	↓	↓
8447-8448	Parameter #128 (File 32)	Read/Write	4224
10241-10242	Parameter #1 (File 40)	Read/Write	5121
↓	↓	↓	↓
10495-10496	Parameter #128 (File 40)	Read/Write	5248
12289-12290	Parameter #1 (File 48)	Read/Write	6145
↓	↓	↓	↓
12543-12544	Parameter #128 (File 48)	Read/Write	6272
14337-14338	Parameter #1 (File 56)	Read/Write	7169
↓	↓	↓	↓
14591-14592	Parameter #128 (File 56)	Read/Write	7296
14801-14802	CO-MUX – SP1	Read/Write	7401
14803-14804	CO-MUX – SP2	Read/Write	7402
14805-14806	CO-MUX – SP3	Read/Write	7403
14807-14808	CO-MUX – T1	Read/Write	7404
14809-14810	CO-MUX – T2	Read/Write	7405
14811-14812	CO-MUX – T3	Read/Write	7406
14813-14814	CO-MUX – T4	Read/Write	7407
14815-14816	CO-MUX – T5	Read/Write	7408
14817-14818	CO-MUX – T6	Read/Write	7409
14819-14820	CO-MUX – T7	Read/Write	7410
14821-14822	CO-MUX – T8	Read/Write	7411
14823-14824	CO-MUX – T9	Read/Write	7412
14825-14826	CO-MUX – T10	Read/Write	7413
14827-14828	CO-MUX – Use A.Out	Read/Write	7414
14829-14830	CO-MUX – SP4	Read/Write	7415
14831-14832	CO-MUX – A.Out Low	Read/Write	7416
14833-14834	CO-MUX – A.Out High	Read/Write	7417
14835-14836	CO-MUX – Work OffLine	Read/Write	7418
14841-14842	CO-MUX – Sensor # 1	Read/Write	7421
14843-14844	CO-MUX – Sensor # 2	Read/Write	7422
↓	↓	↓	↓
14899-14900	CO-MUX – Sensor # 30	Read/Write	7450
14901-14902	CO-MUX – Dout # 1	Read/Write	7451

<b>MODBUS Register</b>	<b>Field Description</b>	<b>Type</b>	<b>ITEM No (Uniart)</b>
↓	↓	↓	↓
14915-14916	CO-MUX – Dout # 8	Read/Write	7458
14917-14918	CO-MUX – Aout # 1	Read/Write	7459
↓	↓	↓	↓
14931-14932	CO-MUX – Aout # 8	Read/Write	7466
16393-16394	Ignore Offset for Parameters (8000)	Read/Write	8197
16395-16396	Web Authentication (0=Free) (User : admin)	Read/Write	8198
16397-16398	Program Number (110..) (SB_ARM)	Read/Write	8199
16401-16402	Analog Input #1	Read	8201
↓	↓	↓	↓
16415-16416	Analog Input #8	Read	8208
16417-16418	Analog Input #1 Force Status	Read	8209
↓	↓	↓	↓
16431-16432	Analog Input #8 Force Status	Read	8216
16433-16434	Digital Input #1 (AIn 1)	Read	8217
↓	↓	↓	↓
16447-16448	Digital Input #8 (AIn 8)	Read	8224
16449-16450	Digital Input #1 (AIn 1) Force Status	Read	8225
↓	↓	↓	↓
16463-16464	Digital Input #8 (AIn 8) Force Status	Read	8232
16465-16466	Analog Output #1	Read/Write	8233
↓	↓	↓	↓
16479-16480	Analog Output #8	Read/Write	8240
16481-16482	Analog Output #1 Force Status	Read/Write	8241
↓	↓	↓	↓
16495-16496	Analog Output #8 Force Status	Read/Write	8248
16497-16498	Digital Output #1	Read/Write	8249
↓	↓	↓	↓
16511-16512	Digital Output #8	Read/Write	8256
16513-16514	Digital Output #1 Force Status	Read/Write	8257

<b>MODBUS Register</b>	<b>Field Description</b>	<b>Type</b>	<b>ITEM No (Uniart)</b>
↓	↓	↓	↓
16527-16528	Digital Output #8 Force Status	Read/Write	8264
16529-16530	Analog Input #1 (Ofset)	Read/Write	8265
↓	↓	↓	↓
16543-16544	Analog Input #8 (Ofset)	Read/Write	8272
16545-16546	Analog Input #1 (Mode Of Meassure)	Read/Write	8273
↓	↓	↓	↓
16559-16560	Analog Input #8 (Mode Of Meassure)	Read/Write	8280
16561-16562	Analog Input #1 (Constant A)	Read/Write	8281
↓	↓	↓	↓
16575-16576	Analog Input #8 (Constant A)	Read/Write	8288
16577-16578	Analog Input #1 (Constant B)	Read/Write	8289
↓	↓	↓	↓
16591-16592	Analog Input #8 (Constant B)	Read/Write	8296
16601-16602	Alarm #1	Read	8301
↓	↓	↓	↓
16727-16728	Alarm #64	Read	8364
16741-16742	Digital Input #1	Read	8371
↓	↓	↓	↓
16755-16756	Digital Input #8	Read	8378
16757-16758	Digital Input #1 Force Status	Read	8379
↓	↓	↓	↓
16771-16772	Digital Input #8 Force Status	Read	8386
16801-16802	SST #1 – Start 1 (Mon-Fri) ၂-၃	Read/Write	8401
16803-16804	SST #1 – Stop 1 (Mon-Fri) ၂-၃	Read/Write	8402
16805-16806	SST #1 – Start 2 (Mon-Fri) ၂-၃	Read/Write	8403

<b>MODBUS Register</b>	<b>Field Description</b>	<b>Type</b>	<b>ITEM No (Uniart)</b>
16807-16808	SST #1 – Stop 2 (Mon-Fri) 17-18	Read/Write	8404
16809-16810	SST #1 – Start 1 (Saturday) 19	Read/Write	8405
16811-16812	SST #1 – Stop 1 (Saturday) 19	Read/Write	8406
16813-16814	SST #1 – Start 2 (Saturday) 19	Read/Write	8407
16815-16816	SST #1 – Stop 2 (Saturday) 19	Read/Write	8408
16817-16818	SST #1 – Start 1 (Sunday) 20	Read/Write	8409
16819-16820	SST #1 – Stop 1 (Sunday) 20	Read/Write	8410
16821-16822	SST #1 – Start 2 (Sunday) 20	Read/Write	8411
16823-16824	SST #1 – Stop 2 (Sunday) 20	Read/Write	8412
16825-16826	SST #2 – Start 1 (Mon-Fri) 17-18	Read/Write	8413
↓	↓	↓	↓
16847-16848	SST #2 – Stop 2 (Sunday) 20	Read/Write	8424
↓	↓	↓	↓
16991-16992	SST #8 – Stop 2 (Sunday) 20	Read/Write	8496
17001-17002	SST #1 – Status	Read	8501
17003-17004	SST #2 – Status	Read	8502
↓	↓	↓	↓
17015-17016	SST #8 – Status	Read	8508
17021-17022	Clock : Seconds	Read/Write	8511
17023-17024	Clock : Minutes	Read/Write	8512
17025-17026	Clock : Hour	Read/Write	8513
17027-17028	Clock : Week Day (1-7)	Read/Write	8514
17029-17030	Clock : Day	Read/Write	8515
17031-17032	Clock : Month	Read/Write	8516
17033-17034	Clock : Year (20xx)	Read/Write	8517
17035-17036	Clock : Time (Win Format)	Read	8518
17201-17202	Filter Avarage for Ain #1	Read/Write	8601
↓	↓	↓	↓
17215-17216	Filter Avarage for Ain #8	Read/Write	8608
17401-17402	Digital Input #1	Read/Write	8701



<b>MODBUS Register</b>	<b>Field Description</b>	<b>Type</b>	<b>ITEM No (Uniart)</b>
↓	↓	↓	↓
17431-17432	Digital Input #16	Read/Write	8716
17465-17466	Digital Input #1 Force Status	Read/Write	8733
↓	↓	↓	↓
17495-17496	Digital Input #16 Force Status	Read/Write	8748
17529-17530	Digital Output #1	Read/Write	8765
↓	↓	↓	↓
17559-17560	Digital Output #16	Read/Write	8780
17593-17594	Digital Output #1 Force Status	Read/Write	8797
↓	↓	↓	↓
17623-17624	Digital Output #16 Force Status	Read/Write	8812
17991-17992	Answer 12.34	Read	8996
17993-17994	Version Num (*100)	Read	8997
17995-17996	Version Num	Read	8998
17997-17998	User Parameter (Technical)	Read/Write	8999

Table 1-4 **Registers Table**

## **What New :**

- 24.10.2006 :** First
- 30.01.2007 :** Split System A From B
- 28.05.2008 :** Fix D.In Regiters + Add Avarage
- 08.07.2008 :** Add Regs 996-998 For debug
- 12.01.2010 :** Add Support To Files (Offset 8000)