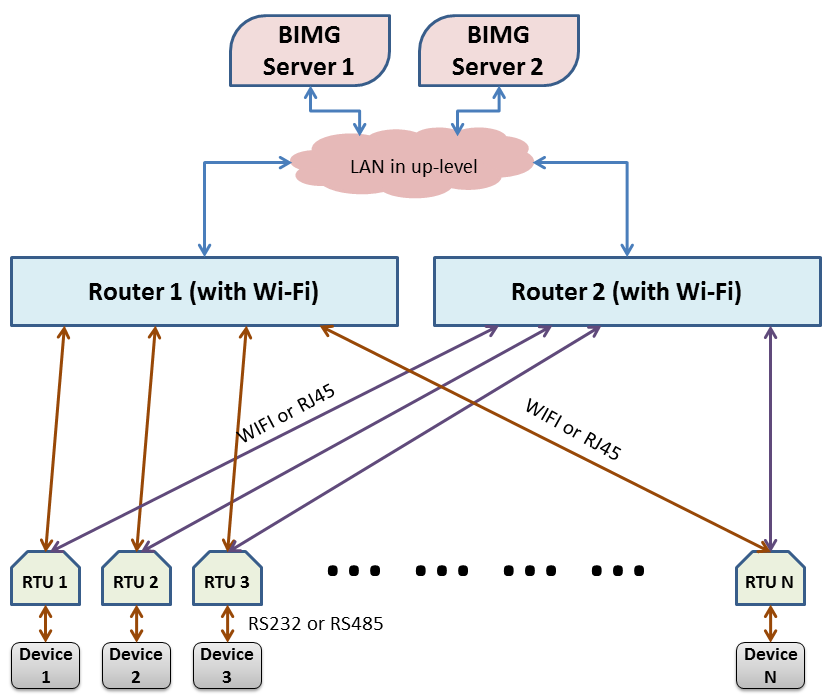
**The sensor network integration information for Smart-meters and Switches using in the SinBerBEST BIMG Test-Bedding**

2014.09.15

There are 12 existing branches in the SinBerBEST BIMG Test-Bedding, which are equipped with smart-meters (9 DC-meters and 3 AC meters). These branches are going to be remotely controlled by means of relays. Therefore, there are also 12 + 2 switches/AC connectors controlling and monitoring units (2 plus are for AC/DC grid power supply selecting relay control). To construct a distributing control system, each branch’s data collecting and relay control is implemented by a Remote-Terminal-Unit (RTU) developed on Raspberry Pi. (The control network layout is shown in the following Figure).



**Figure 1. Layout of SinBerBEST BIMG Test-Bedding Control Network**

These RTUs communicates with BIMG server(s) in using the new technology—Websocket, so that the control network achieves a fast & steady data-exchange in real-time communication. Table 1 shows a summary to the sensor network integration for the SinBerBEST BIMG Test-Bedding.

Table 1. Main Information for integration.

|  |  |  |
| --- | --- | --- |
| Items | Parameter | Description |
| **Communication Media** | Wi-Fi and/or RJ45 | Multi-media offer a flexible and no-single-failure communication. |
| **Communication Paths** | Linking to BIMG servers by means of routers/Wi-Fi routers. | Web-Socket Technology |
| **Location** | The dry-lab |  |
| **Data-collecting rate** | Normal data (e.g. for voltage display refreshment)—less than 1 s;  Energy data—60 s; | All of the refresh-time-interval can be re-setting to a more proper value by BIMG server. |
| **Switch(Relay)-status collecting** | On server request or being triggered by switch-statue change. |  |
| **\*Communication Protocol** | Modbus TCP IP and Modbus-RTU | Please see the relevant Modbus Standards or reference appendix 1 & 2 |

**Appendix 1: Smart-meter SENTRON PAC3200**

The smart-meters deployed in the SinBerBEST BIMG Test-Bedding AC bus side are SENTRON PAC3200, which are 380 V three-phase power monitoring device using for AC grid.

At present, the BIMG server communicates with PAC3200 by means of Ethernet (LAN, i.e. RJ45). The control commands of PAC3200 are a sub-set of Mod-Bus TCP standard.

1. Communication work principal

When PAC3200 is power on, it starts up with self-initialization process. After that, it automatically goes into slave mode and waits for the BIMG server sending request command. Each time the PAC3200 receiving a correct request, it returns a response to the BIMG server. So the data collecting rate depends on the BIMG server rather than PAC3200. Usually, the data collecting rate is no more than 1.5 per (request & response round) /second. If the data collecting rate is too fast for a PAC3200, it may cause jam inside the PAC3200.

1. The communication protocol of PAC3200 using in the SinBetBEST BIMG Test-Bedding

When PAC3200 is powered on, it starts up with self-initialization process. After that process, it automatically goes into slave mode and waits for the BIMG server, which is working in master mode, sending request command. Each time the PAC3200 receiving a correct request, it returns a response. So the data collecting rate depends on the BIMG server rather than PAC3200. Usually, the data collecting rate is no more than 1.5 per request & response round /second. If the data collecting rate is too fast for a PAC3200, it may cause a jam inside the PAC3200.

At present, the SinBerBEST BIMG Test-Bedding employs a very basic communication protocol of PAC3200 and the protocol will be extended later. There are two query commands only:

**Query command 1: Normal data query**

The request command in Java format is:

byte[] requestTimerRefreshArray = new byte[]{86,80,0,0,0,6,(byte)255,03,0,01,0,70}; (detailed request command explanation please see the document of Mod-Bus standard)

This request command is to retrieve the data in the following table (note: the response of PAC3200 is a byte[]; the table only lists the useful data in the byte[]):

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Offset | Bytes | Name | Format | Unit | Value range | Access |
| 1\*2+7 | 2\*2 | Voltage Va-n | Float | V | - | R |
| 3\*2+7 | 2\*2 | Voltage Vb-n | Float | V | - | R |
| 5\*2+7 | 2\*2 | Voltage Vc-n | Float | V | - | R |
| 7\*2+7 | 2\*2 | Voltage Va-b | Float | V | - | R |
| 9\*2+7 | 2\*2 | Voltage Vb-c | Float | V | - | R |
| 11\*2+7 | 2\*2 | Voltage Vc-a | Float | V | - | R |
| 13\*2+7 | 2\*2 | Current a | Float | A | - | R |
| 15\*2+7 | 2\*2 | Current b | Float | A | - | R |
| 17\*2+7 | 2\*2 | Current c | Float | A | - | R |
| 19\*2+7 | 2\*2 | Apparent Power a | Float | VA | - | R |
| 21\*2+7 | 2\*2 | Apparent Power b | Float | VA | - | R |
| 23\*2+7 | 2\*2 | Apparent Power c | Float | VA | - | R |
| 25\*2+7 | 2\*2 | Active Power a | Float | W | - | R |
| 27\*2+7 | 2\*2 | Active Power b | Float | W | - | R |
| 29\*2+7 | 2\*2 | Active Power c | Float | W | - | R |
| 31\*2+7 | 2\*2 | Reactive Power a | Float | var | - | R |
| 33\*2+7 | 2\*2 | Reactive Power b | Float | var | - | R |
| 35\*2+7 | 2\*2 | Reactive Power c | Float | var | - | R |
| 37\*2+7 | 2\*2 | Power Factor a | Float | - | 0 ... 1 | R |
| 39\*2+7 | 2\*2 | Power Factor b | Float | - | 0 ... 1 | R |
| 41\*2+7 | 2\*2 | Power Factor c | Float | - | 0 ... 1 | R |
| 43\*2+7 | 2\*2 | THD-R Voltage a | Float | % | 0 ... 100 | R |
| 45\*2+7 | 2\*2 | THD-R Voltage b | Float | % | 0 ... 100 | R |
| 47\*2+7 | 2\*2 | THD-R Voltage c | Float | % | 0 ... 100 | R |
| 49\*2+7 | 2\*2 | THD-R Current a | Float | % | 0 ... 100 | R |
| 51\*2+7 | 2\*2 | THD-R Current b | Float | % | 0 ... 100 | R |
| 53\*2+7 | 2\*2 | THD-R Current c | Float | % | 0 ... 100 | R |
| 55\*2+7 | 2\*2 | Frequency | Float | Hz | 45 ... 65 | R |
| 57\*2+7 | 2\*2 | Average Voltage Vph-n | Float | V | - | R |
| 59\*2+7 | 2\*2 | Average Voltage Vph-ph | Float | V | - | R |
| 61\*2+7 | 2\*2 | Average Current | Float | A | - | R |
| 63\*2+7 | 2\*2 | Total Apparent Power | Float | VA | - | R |
| 65\*2+7 | 2\*2 | Total Active Power | Float | W | - | R |
| 67\*2+7 | 2\*2 | Total Reactive Power | Float | var | - | R |
| 69\*2+7 | 2\*2 | Total Power Factor | Float |  | - | R |

**Query command 2: Energy data query**

The request in Java format is:

byte[] requestTimerEnergyArray = new byte[]{87,87,0,0,0,6,(byte)255,03,3,33,0,40}; (detailed request command explanation please see the document of Mod-Bus standard)

This request command is to retrieve the data in the following table (note: the response of PAC3200 is a byte[]; the table only lists the useful data in the byte[]):

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Offset | Bytes | Name | Format | Unit | Value range | Access |
| 2\*(801-801)+9 | 4\*2 | Active Energy Import Tariff 1 | Double | Wh | Overflow 1.0e+12 | RW |
| 2\*(805-801)+9 | 4\*2 | Active Energy Import Tariff 2 | Double | Wh | Overflow 1.0e+12 | RW |
| 2\*(809-801)+9 | 4\*2 | Active Energy Export Tariff 1 | Double | Wh | Overflow 1.0e+12 | RW |
| 2\*(813-801)+9 | 4\*2 | Active Energy Export Tariff 2 | Double | Wh | Overflow 1.0e+12 | RW |
| 2\*(817-801)+9 | 4\*2 | Reactive Energy Import Tariff 1 | Double | varh | Overflow 1.0e+12 | RW |
| 2\*(821-801)+9 | 4\*2 | Reactive Energy Import Tariff 2 | Double | varh | Overflow 1.0e+12 | RW |
| 2\*(825-801)+9 | 4\*2 | Reactive Energy Export Tariff 1 | Double | varh | Overflow 1.0e+12 | RW |
| 2\*(829-801)+9 | 4\*2 | Reactive Energy Export Tariff 2 | Double | varh | Overflow 1.0e+12 | RW |
| 2\*(833-801)+9 | 4\*2 | Apparent Energy Tariff 1 | Double | VAh | Overflow 1.0e+12 | RW |
| 2\*(837-801)+9 | 4\*2 | Apparent Energy Tariff 2 | Double | VAh | Overflow 1.0e+12 | RW |

To better understanding the above request commands of PAC3200, the available measured variables of SENTRON PAC3200 are listed in table below:

Available measured variables of SENTRON PAC3200

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Offset | Number of registers | Name | Format | Unit | Value range | Access |
| 1 | 2 | Voltage Va-n | Float | V | - | R |
| 3 | 2 | Voltage Vb-n | Float | V | - | R |
| 5 | 2 | Voltage Vc-n | Float | V | - | R |
| 7 | 2 | Voltage Va-b | Float | V | - | R |
| 9 | 2 | Voltage Vb-c | Float | V | - | R |
| 11 | 2 | Voltage Vc-a | Float | V | - | R |
| 13 | 2 | Current a | Float | A | - | R |
| 15 | 2 | Current b | Float | A | - | R |
| 17 | 2 | Current c | Float | A | - | R |
| 19 | 2 | Apparent Power a | Float | VA | - | R |
| 21 | 2 | Apparent Power b | Float | VA | - | R |
| 23 | 2 | Apparent Power c | Float | VA | - | R |
| 25 | 2 | Active Power a | Float | W | - | R |
| 27 | 2 | Active Power b | Float | W | - | R |
| 29 | 2 | Active Power c | Float | W | - | R |
| 31 | 2 | Reactive Power a | Float | var | - | R |
| 33 | 2 | Reactive Power b | Float | var | - | R |
| 35 | 2 | Reactive Power c | Float | var | - | R |
| 37 | 2 | Power Factor a | Float | - | 0 ... 1 | R |
| 39 | 2 | Power Factor b | Float | - | 0 ... 1 | R |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 41 | 2 | Power Factor c | Float | - | 0 ... 1 | R |
| 43 | 2 | THD-R Voltage a | Float | % | 0 ... 100 | R |
| 45 | 2 | THD-R Voltage b | Float | % | 0 ... 100 | R |
| 47 | 2 | THD-R Voltage c | Float | % | 0 ... 100 | R |
| 49 | 2 | THD-R Current a | Float | % | 0 ... 100 | R |
| 51 | 2 | THD-R Current b | Float | % | 0 ... 100 | R |
| 53 | 2 | THD-R Current c | Float | % | 0 ... 100 | R |
| 55 | 2 | Frequency | Float | Hz | 45 ... 65 | R |
| 57 | 2 | Average Voltage Vph-n | Float | V | - | R |
| 59 | 2 | Average Voltage Vph-ph | Float | V | - | R |
| 61 | 2 | Average Current | Float | A | - | R |
| 63 | 2 | Total Apparent Power | Float | VA | - | R |
| 65 | 2 | Total Active Power | Float | W | - | R |
| 67 | 2 | Total Reactive Power | Float | var | - | R |
| 69 | 2 | Total Power Factor | Float |  | - | R |
| 71 | 2 | Amplitude Unbalance - Voltage | Float | % | 0 ... 100 | R |
| 73 | 2 | Amplitude Unbalance - Current | Float | % | 0 ... 200 | R |
| 75 | 2 | Maximum Voltage Va-n | Float | V | - | R |
| 77 | 2 | Maximum Voltage Vb-n | Float | V | - | R |
| 79 | 2 | Maximum Voltage Vc-n | Float | V | - | R |
| 81 | 2 | Max. Voltage Va-b | Float | V | - | R |
| 83 | 2 | Max. Voltage Vb-c | Float | V | - | R |
| 85 | 2 | Max. Voltage Vc-a | Float | V | - | R |
| 87 | 2 | Maximum Current a | Float | A | - | R |
| 89 | 2 | Maximum Current b | Float | A | - | R |
| 91 | 2 | Maximum Current c | Float | A | - | R |
| 93 | 2 | Maximum Apparent Power a | Float | VA | - | R |
| 95 | 2 | Maximum Apparent Power b | Float | VA | - | R |
| 97 | 2 | Maximum Apparent Power c | Float | VA | - | R |
| 99 | 2 | Maximum Active Power a | Float | W | - | R |
| 101 | 2 | Maximum Active Power b | Float | W | - | R |
| 103 | 2 | Maximum Active Power c | Float | W | - | R |
| 105 | 2 | Maximum Reactive Power a | Float | var | - | R |
| 107 | 2 | Maximum Reactive Power b | Float | var | - | R |
| 109 | 2 | Maximum Reactive Power c | Float | var | - | R |
| 111 | 2 | Maximum Power Factor a | Float |  | 0 ... 1 | R |
| 113 | 2 | Maximum Power Factor b | Float |  | 0 ... 1 | R |
| 115 | 2 | Maximum Power Factor c | Float |  | 0 ... 1 | R |
| 117 | 2 | Maximum THD-R Voltage a | Float | % | 0 ... 100 | R |
| 119 | 2 | Maximum THD-R Voltage b | Float | % | 0 ... 100 | R |
| 121 | 2 | Maximum THD-R Voltage c | Float | % | 0 ... 100 | R |
| 123 | 2 | Maximum THD-R Current a | Float | % | 0 ... 100 | R |
| 125 | 2 | Maximum THD-R Current b | Float | % | 0 ... 100 | R |
| 127 | 2 | Maximum THD-R Current c | Float | % | 0 ... 100 | R |
| 129 | 2 | Max. Frequency | Float | Hz | 45 ... 65 | R |
| 131 | 2 | Max. Average Voltage Vph-n | Float | V | - | R |
| 133 | 2 | Max. Average Voltage Vph-ph | Float | V | - | R |
| 135 | 2 | Max. Average Current | Float | A | - | R |
| 137 | 2 | Max. Total Apparent Power | Float | VA | - | R |
| 139 | 2 | Max. Total Active Power | Float | W | - | R |
| 141 | 2 | Max. Total Reactive Power | Float | var | - | R |
| 143 | 2 | Maximum Total Power Factor | Float |  | - | R |
| 145 | 2 | Minimum Voltage Va-n | Float | V | - | R |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 147 | 2 | Minimum Voltage Vb-n | Float | V | - | R |
| 149 | 2 | Minimum Voltage Vc-n | Float | V | - | R |
| 151 | 2 | Min. Voltage Va-b | Float | V | - | R |
| 153 | 2 | Min. Voltage Vb-c | Float | V | - | R |
| 155 | 2 | Min. Voltage Vc-a | Float | V | - | R |
| 157 | 2 | Minimum Current a | Float | A | - | R |
| 159 | 2 | Minimum Current b | Float | A | - | R |
| 161 | 2 | Minimum Current c | Float | A | - | R |
| 163 | 2 | Minimum Apparent Power a | Float | VA | - | R |
| 165 | 2 | Minimum Apparent Power b | Float | VA | - | R |
| 167 | 2 | Minimum Apparent Power c | Float | VA | - | R |
| 169 | 2 | Minimum Active Power a | Float | W | - | R |
| 171 | 2 | Minimum Active Power b | Float | W | - | R |
| 173 | 2 | Minimum Active Power c | Float | W | - | R |
| 175 | 2 | Minimum Reactive Power a | Float | var | - | R |
| 177 | 2 | Minimum Reactive Power b | Float | var | - | R |
| 179 | 2 | Minimum Reactive Power c | Float | var | - | R |
| 181 | 2 | Minimum Power Factor a | Float | - | 0 ... 1 | R |
| 183 | 2 | Minimum Power Factor b | Float | - | 0 ... 1 | R |
| 185 | 2 | Minimum Power Factor c | Float | - | 0 ... 1 | R |
| 187 | 2 | Min. Frequency | Float | Hz | 45 ... 65 | R |
| 189 | 2 | Min. Average Voltage Vph-n | Float | V | - | R |
| 191 | 2 | Min. Average Voltage Vph-ph | Float | V | - | R |
| 193 | 2 | Min. Average Current | Float | A | - | R |
| 195 | 2 | Min. Total Apparent Power | Float | VA | - | R |
| 197 | 2 | Min. Total Active Power | Float | W | - | R |
| 199 | 2 | Min. Total Reactive Power | Float | var | - | R |
| 201 | 2 | Minimum Total Power Factor | Float | var | - | R |
| 203 | 2 | Limit Violations\* | Unsigned long | - | Byte 3 Bit 0 Limit 0 | R |
| 205 | 2 | Device Diagnostics and Device Status\* | Unsigned long | - | Byte 0 System status | R |
| 207 | 2 | Status of the digital outputs\* | Unsigned long | - | Byte 3 Bit 0 Output 0 | R |
| 209 | 2 | Status of the digital inputs\* | Unsigned long | - | Byte 3 Bit 0 Input 0 | R |
| 211 | 2 | Active Tariff | Unsigned long |  | 1. = Tariff 1 2. = Tariff 2 | R |
| 213 | 2 | Working hours counter | Unsigned long | s | 0 ...999999999 | RW |
| 215 | 2 | Universal counter | Unsigned long | - | 0 ...999999999 | RW |
| 217 | 2 | Relevant Parameter Changes Counter | Unsigned long | - | - | R |
| 219 | 2 | Counter All Parameter Changes | Unsigned long | - | - | R |
| 221 | 2 | Counter Limit Violations | - | - | - | R |
| 501 | 2 | Demand Active Power - Import | Float | W | - | R |
| 503 | 2 | Demand Reactive Power - Import | Float | var | - | R |
| 505 | 2 | Demand Active Power - Export | Float | W | - | R |
| 507 | 2 | Demand Reactive Power - Export | Float | var | - | R |
| 509 | 2 | Maximum Active Power Reading during the period | Float | W | - | R |
| 511 | 2 | Minimum Active Power Reading during the period | Float | W | - | R |
| 513 | 2 | Maximum Reactive Power Reading during the period | Float | var | - | R |
| 515 | 2 | Minimum Reactive Power Reading during the period | Float | var | - | R |
| 517 | 2 | Demand Period | Unsigned long | s | - | R |
| 519 | 2 | Time Since Start of the active demand period | Unsigned long | s | - | R |
| 801 | 4 | Active Energy Import Tariff 1 | Double | Wh | Overflow 1.0e+12 | RW |
| 805 | 4 | Active Energy Import Tariff 2 | Double | Wh | Overflow 1.0e+12 | RW |
| 809 | 4 | Active Energy Export Tariff 1 | Double | Wh | Overflow 1.0e+12 | RW |
| 813 | 4 | Active Energy Export Tariff 2 | Double | Wh | Overflow 1.0e+12 | RW |
| 817 | 4 | Reactive Energy Import Tariff 1 | Double | varh | Overflow 1.0e+12 | RW |
| 821 | 4 | Reactive Energy Import Tariff 2 | Double | varh | Overflow 1.0e+12 | RW |
| 825 | 4 | Reactive Energy Export Tariff 1 | Double | varh | Overflow 1.0e+12 | RW |
| 829 | 4 | Reactive Energy Export Tariff 2 | Double | varh | Overflow 1.0e+12 | RW |
| 833 | 4 | Apparent Energy Tariff 1 | Double | VAh | Overflow 1.0e+12 | RW |
| 837 | 4 | Apparent Energy Tariff 2 | Double | VAh | Overflow 1.0e+12 | RW |

Abbr. in the "Access" column Abbreviation

R Read; read access

W Write; write access

RW Read and Write; read and write access

**Appendix 2: DC-meter AcuDC243**

The DC-meters deployed in the SinBerBEST BIMG Test-Bedding DC bus side are DC-meter AcuDC243, which are 380 VDC power monitoring device using for DC grid.

At present, the BIMG server dose not directly link with AcuDC243 but through Raspberry PI. The Raspberry PI connects with the AcuDC243 using its USB/SPI interface and the communication protocol of AcuDC243 in using is the Mod-Bus RTU (a version for serial port communication protocol standard). Mod-Bus RTU is relatively complicated as it needs to deal with the CRC communication error check as well as other issues. However, these tasks have been done in the firmware/software of Raspberry PI.

The BIMG server links with the Raspberry PI by means of Ethernet (LAN, i.e. RJ45) or WiFi. And the communication protocol is derived and simplified based on the Mod-Bus RTU protocol in the AcuDC243 user's manual.1. Data Query

**Query commandl: Normal data query**

The request command in Java format is:

byte[] requestTimerRefreshArray = new byte[]{1,3,2,0,0,20,0,0};

(detailed request command explanation please see the document of Mod-Bus RTU standard)

This request command is to retrieve the data in the following response byte[]:

This request command is to retrieve the data in the following response byte[]:

This request command is to retrieve the data in the following response byte[]:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Offset | Bytes | Name | Format | Unit | Value range | Access |
| 0 | 3 | Reserved (Adr,Fun,CT) | - | - | - | - |
| 3 | 4 | Voltage | Float | V | 5~1200 | R |
| 7 | 4 | Current | Float | A | 20~5000 | R |
| 11 | 4 | Power | Float | kW/100 | - | R |
| 15 | 4 | Reserved | - | - | - | R |
| 19 | 4 | Reserved | - | - | - | R |
| 23 | 2 | Reserved (CRC) |  |  |  |  |

**Query command 2: Energy data query**

The request in Java format is:

byte[] requestTimerEnergyArray = new byte[]{1,3,3,0,0,16,0,0}; (detailed request command explanation please see the document of Mod-Bus RTU standard)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Offset | Bytes | Name | Format | Unit | Value range | Access |
| 0 | 3 | Reserved (Adr,Fun,CT) | - | - | - | - |
| 3 | 4 | Export Energy | Long | kWh/100 | 0~999999999 | R |
| 7 | 4 | Import Energy | Long | kWh/100 | 0~999999999 | R |
| 11 | 4 | Total Energy | Long | kWh/100 | 0~999999999 | R |
| 15 | 4 | Net Energy | Long | kWh/100 | 0~999999999 | R |
| 19 | 2 | Reserved (CRC) |  |  |  |  |

**Query command 3: Time running query**

The request in Java format is:

byte[] requestTimerRunningArray = new byte[]{1,3,2,0x80,0,8,0,0}; (detailed request command explanation please see the document of Mod-Bus RTU standard)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Offset | Bytes | Name | Format | Unit | Value range | Access |
| 0 | 3 | Reserved (Adr,Fun,CT) | - | - | - | - |
| 3 | 4 | Meter Running Time | Long | Hour/100 | 0~999999999 | R |
| 7 | 4 | Load Running Time | Long | Hour/100 | 0~999999999 | R |
| 11 | 2 | Reserved (CRC) |  |  |  |  |

2. Switch Monitoring and Relay Control

Being integrated with the DC-meter AcuDC243, the Raspberry PI also can implement the switch monitoring and relay control function on the DC grid node (or branch) in which the DC-meter AcuDC243 is sited.

**Query commandl: Switch monitoring query**

The request command in Java format is:

byte[] requestSwitchStatusArray = new byte[]{1,2,0,0,0,2,0,0};

(detailed request command explanation please see the document of Mod-Bus RTU standard)

This request command is to retrieve the data in the following response byte[]:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Offset | Bytes | Name | Format | Unit | Value range | Access |
| 0 | 3 | Reserved (Adr,Fun,CT) | - | - | - | - |
| 3 | 1 | Switch Status | bit |  | 1. Switch1 off, Switch2 off; 2. Switch1 on, Switch2 off; 3. Switch1 off, Switch2 on; 4. Switch1 on, Switch2 on; Other values—reserved | R |
| 4 | 2 | Reserved (CRC) |  |  |  |  |

**Query command2: Relay status query**

The request command in Java format is:

byte[] requestRelayStatusArray = new byte[]{1,1,0,0,0,2,0,0};

(detailed request command explanation please see the document of Mod-Bus RTU standard)

This request command is to retrieve the data in the following response byte[]:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Offset | Bytes | Name | Format | Unit | Value range | Access |
| 0 | 3 | Reserved (Adr,Fun,CT) | - | - | - | - |
| 3 | 1 | Relay Status | bit |  | 1. Relay1 off, Relay 2 off; 2. Relay 1 on, Relay 2 off; 3. Relay 1 off, Relay 2 on; 4. Relay 1 on, Relay 2 on; Other values—reserved | R |
| 4 | 2 | Reserved (CRC) |  |  |  |  |

**Control command: Relay control commands**

There are four relay control commands. These are (in Java format)

* Relayl off and Relay2 off:

byte[] requestRelayControlOffOffArray = new byte[]{1,5,0,0,0,0,0,0};

* Relayl on and Relay2 off:

byte[] requestRelayControlOffOffArray = new byte[]{1,5,0,0,0xff,0,0,0};

* Relayl off and Relay2 on:

byte[] requestRelayControlOffOffArray = new byte[]{1,5,0,1,0xff,0,0,0};

* Relayl on and Relay2 on:

byte[] requestRelayControlOffOffArray = new byte[]{1,5,0,1,0xff,0,0,0}; (detailed request command explanation please see the document of Mod-Bus RTU standard)

**Chapter 3: Switch (or AC Contactor) Monitoring and Relay Control**

In the SinBerBEST BIMG Test-Bedding, all of the switches (or AC contactor) and relays control are implemented by Remote Control Unit (RTU) other than directly link with the BIMG server. At present, the RTU is developed on Raspberry PI B which supports communication media in Ethernet (LAN, i.e. RJ45) and WiFi. To simplify design, the communication protocol of the RTU is derived and simplified based on the Mod-Bus RTU, (which is same to the previous chapter).

**Query command1: Switch monitoring query**

The request command in Java format is:

byte[] requestSwitchStatusArray = new byte[]{1,2,0,0,0,2,0,0};

(detailed request command explanation please see the document of Mod-Bus RTU standard) This request command is to retrieve the data in the following response byte[]:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Offset | Bytes | Name | Format | Unit | Value range | Access |
| 0 | 3 | Reserved (Adr,Fun,CT) | - | - | - | - |
| 3 | 1 | Switch Status | bit |  | 1. Switch1 off, Switch2 off; 2. Switch1 on, Switch2 off; 3. Switch1 off, Switch2 on; 4. Switch1 on, Switch2 on; Other values—reserved | R |
| 4 | 2 | Reserved (CRC) |  |  |  |  |

**Query command!: Relay status query**

The request command in Java format is:

byte[] requestRelayStatusArray = new byte[]{1,1,0,0,0,2,0,0};

(detailed request command explanation please see the document of Mod-Bus RTU standard)

This request command is to retrieve the data in the following response byte[]:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Offset | Bytes | Name | Format | Unit | Value range | Access |
| 0 | 3 | Reserved (Adr,Fun,CT) | - | - | - | - |
| 3 | 1 | Relay Status | bit |  | 1. Relay1 off, Relay 2 off; 2. Relay 1 on, Relay 2 off; 3. Relay 1 off, Relay 2 on; 4. Relay 1 on, Relay 2 on; Other values—reserved | R |
| 4 | 2 | Reserved (CRC) |  |  |  |  |

**Control command: Relay control commands**

There are four relay control commands. These are (in Java format)

* Relayl off and Relay2 off:

byte[] requestRelayControlOffOffArray = new byte[]{1,5,0,0,0,0,0,0};

* Relayl on and Relay2 off:

byte[] requestRelayControlOffOffArray = new byte[]{1,5,0,0,0xff,0,0,0};

* Relayl off and Relay2 on:

byte[] requestRelayControlOffOffArray = new byte[]{1,5,0,1,0,0,0,0};

* Relayl on and Relay2 on:

byte[] requestRelayControlOffOffArray = new byte[]{1,5,0,1,0xff,0,0,0}; (detailed request command explanation please see the document of Mod-Bus RTU standard)