KinCony KCSv3 protocol – MQTT

Note: This protocol document use for KinCony ESP32-S3 smart controller:

you need to download KCS v3 firmware to ESP32-S3 firstly.

Different board will have different channel of digital output, digital input , ADC, DAC,IR, RF. So the protocol is same , just according to the hardware resource to set channel number.

Command topic format: Board model/UID/SET
State topic format: Board model/UID/STATE

If using KC868-A64:
Command topic: `KC868_A64/B48A0A404664/SET`
State topic: `KC868_A64/B48A0A404664/STATE`

1. Set ON/OFF of one digital output channel

   send: {"output64":{"value":true}} means: turn ON output64
   send: {"output64":{"value":false}} means: turn OFF output64

2. Feedback STATE
   If board first time connect to mqtt broker, will auto feedback all state of board.
   If digital input or output STATE changed by any way, will auto feedback mqtt message.

   For example:

   {"output1":{"value":true}} output1 is ON
   {"output2":{"value":false}} output2 is OFF
   {"input3":{"value":true}} input3 is trigger
   {"input4":{"value":false}} input4 is not trigger

   If you send control output command, then will feedback all STATE of board together, such as:

   {{"output1":{"value":true}},{"output2":{"value":false}},·····{"input1":{"value":true}},{"input2":{"value":true}},·····{"adc1":{"value":2610}},·····{"dac1":{"value":10}},·····}

3. Set DAC value
   Send: {"dac1":{"value":58}}

   Set DAC1 value=58, range is 0-255

4. Digital output ALL ON/OFF
   send: {"all_outputs":{"value":true}}
   turn ON all output

   send: {"all_outputs":{"value":false}}

turn OFF all output

true=ALL ON，false=ALL OFF

5. Read board all data
   send：{"get_datas":{"value":true}}

   feedback all data:
   {{"output1":{"value":true}},{"output2":{"value":false}},·····{"input1":{"value":true}},{"input2":{"value":true}},·····{"adc1":{"value":2610}},·····{"dac1":{"value":10}},·····}

6. Set ON/OFF of multi digital output channel
   send：{{"output1":{"value":true}},{"output2":{"value":false}}} means: turn ON output1 and turn OFF output2

   feedback all STATE
   {{"output1":{"value":true}},{"output2":{"value":false}},·····{"input1":{"value":true}},{"input2":{"value":true}},·····{"adc1":{"value":2610}},·····{"dac1":{"value":10}},·····}

7. **Learn IR:** // begin learn IR signal ID=1, send by IR tube-2 (for example: AG8 controller have 8 channel IR send tube)
   send:
   ```
   {
     "learn_ir": {
       "value": 1,
       "tx_channel": 2
     }
   }
   ```

   Send IR: // send IR signal that ID=1
   send:
   ```
   {
     "run_ir": {
     "value": 1
       }
   }
   ```

   Delete IR: // delete learned IR signal that ID=1
   send:
   ```
   {
     "del_ir": {
     "value": 1
       }
   }
   ```

   IR triggered and auto feedback: // trigger when value=1 (ONLY use for NEC and RC5 IR code)
   receive:
   // trigger begin
   ```
   {
   ```

```
    "ir1": {
    "value": 1
        }
  }

  // trigger end
  {
    "ir1": {
    "value": 0
        }
 }
```

8. Send one RF signal have learned
   send: {"run_rf":{"value":1}} means: send RF signal that id=1

   feedback: {"run_rf1":{"value":success}}

9. Set beep output

   send: {"set_beep":{"value":1}} means: beep ON
   send: {"set_beep":{"value":0}} means: beep OFF

   feedback: {" set_beep":{"value":success}}

10. SIM7600 4G module make CALL and SMS functions:

Making a Call

```
{
  "run_call": {
    "phone": "+8612345678901"
  }
}
```

Successful Response:

```
{
  "run_call": {
    "value": "success"
  }
}
```

Failed Response:
Possible reasons: GPRS is not initialized, or the phone number is not set.

```
{
  "run_call": {
    "value": "error"
  }
}
```

Sending an SMS

```
{
   "run_sms": {
      "content": "sms content example",
      "phone": "+8612345678901"
   }
}
```

Successful Response:

```
{
   "run_sms": {
      "value": "success"
   }
}
```

Failed Response:
Possible reasons: GPRS is not initialized, the phone number is not set, or the SMS content is not set.

```
{
   "run_sms": {
      "value": "error"
   }
}
```

MQTT STATE Reporting Received SMS and Incoming Calls

Incoming Call

```
{
   "call_ring": {
      "phone": "+8612345678901"
   }
}
```

Received SMS

```
{
   "sms_recv": {
      "phone": "+8612345678901",
      "content": "example"
   }
}
```

For N series energy meter

For example:

{"C6_51":0.13,"C6_52":0,"C6_53":0,"C6_54":0,"C6_55":0,"C6_56":0,"C6_57":0,"C6_58":0,"C6_59":0,"C6_60":0,"Voltage6":220,"P6_51":21.23,"P6_52":0,"P6_53":0,"P6_54":0,"P6_55":0,"P6_56":0,"P6_57":0,"P6_58":0,"P6_59":0,"P6_60":0,"E6_51":15,"E6_52":2,"E6_53":0,"E6_54":0,"E6_55":0,"E6_56":0,"E6_57":0,"E6_58":0,"E6_59":0,"E6_60":0,"E6_Sum":17,"Frequency6":50,"TPS6_1":25,"TPS6_2":0}

"C6_51":0.13
C: current
6: sixth energy chip of BL0910
0.13: current 0.13A

“Voltage6":220: voltage channel6 is AC220V

“P6_51":21.23 channel 51 power is 21.23W

“E6_51":15 channel 51 energy meter is 15kwh

"E6_Sum":17

sixth energy chip of BL0910
channel51+channel52+channel53+channel54+channel55+channel56+channel57+channel58+channel59+channel60
=17kwh

"Frequency6":50 voltage 6’s power frequency is 50Hz

TPS6_1 and TPS6_2 are BL0910 chip’s temperature, usually not need to use it.