# AirTouch 5
# Communication Protocol

Polyaire Pty Ltd

| version | data | content | Contributors |
|---------|------|---------|--------------|
| V1.0 | 7/9/2022 | Create | Vincent |
| V1.1 | 5/10/2022 | Added Intelligent Auto fan speed | Vincent |
| V1.2 | 27/6/2024 | Update 4.a.ii Zone status message<br>Update 4.a.iv AC status message | Vincent |

# Contents

# 1.     Overview

AirTouch 5 allows connection through TCP to control the device. It supports querying the status of AC and zones and controlling the AC and zones.

Clients and AirTouch 5 communicate control or status messages by sending packets.

# 2.　　Connection

Join AirTouch 5 console to local WiFi network.

Connect to AirTouch 5 console at port 9005 by TCP protocol. If there are two consoles, connect to the master console. If failed, try to connect to the other one.

Page in "Settings" -> "Installer" -> "General" shows which console is the master(only for two consoles).

To see the IP address of the console, go to "Settings" -> "Options", click the wifi button. Click "SEE MORE" to go to Android wifi setting page. Click "Wi-Fi", then click the connected wifi name  to go to the Network details page. IP address is in the "Advanced" section.

## a. Discovery

AirTouch 5 supports UDP broadcast to discover devices.

Broadcast "**::REQUEST-POLYAIRE-AIRTOUCH-DEVICE-INFO:;**" to port 49005 to discover AirTouch 5 devices on the local network.

AirTouch 5 will response a UDP message to 49005.

Message format is: [IP],[ConsoleID],AirTouch5,[AirTouch ID],[Device Name]

For hubs that integrate home smart devices, it is recommended this message be used only during setup. Fixing the IP of the device in the router will increase stability.

# 3. Package Format

A package has following components:

- Header (4 bytes)
- Address (2 bytes)
- Message id (1 byte)
- Message type (1 byte)
- Data length (2 bytes)
- Data
- CRC16 check bytes (2 bytes)

## a. Header

Header is always 0x55 0x55 0x55 0xAA.

## b. Address

Address should be 0x80 0xb0 or 0x90 0xb0 (for Extended message) when sending to AirTouch. When receiving from AirTouch, last byte of address will be 0x80 or 0x90/0x91 (for Extended message).

## c. Message id

When sending message to AirTouch, message id can be any data. The response message should have the same message id.

## d. Message type

There are two message types: 0xC0 – control command and status message, 0x1F – extended message.

Ignore any other received types.

## e. Data length

Data length is the length of actual data. The first byte is the high byte, the second byte is the low byte.

## f. Data

See section 4 Messages for details.

## g. Check bytes

The algorithm of checksum is CRC16 MODBUS. Use all the package bytes except the header(Address, Message id, Message type, Data length, Data).

## h. Redundant bytes in package

To prevent the package bytes from containing the same bytes as header, a 00 is inserted after every three consecutive 0x55s in the package. The inserted 00 is redundant bytes. Redundant bytes do not participate in check calculation.

# 4.    Messages

## a. Control command and status message (0xC0)

This message contains sub message type (Zone control, Zone status, AC control and AC status), data length detail and sub data.

First 8 bytes are the sub message type and data length details.

| Byte1 | Sub message type |
|-------|------------------|
| Byte2 | Keep 0 |
| Byte3 | Normal data length |
| Byte4 | |
| Byte5 | Each repeat data length |
| Byte6 | |
| Byte7 | Repeat data count |
| Byte8 | |

For one (0xC0) message:

Data length (3.e.) = 8 + Normal data length + repeat data length * repeat data count

Sub data length = Normal data length + repeat data length * repeat data count = Data length - 8

### i. Zone control (0x20)

Zone control messages are to control all zones. Each message to AirTouch is to control one or more specific zones.

No normal data (byte3 byte4: 0).

Each repeat data (4 bytes) controls one specific zone. (byte5 byte6: 0x00 0x04).

| Byte in repeat data | | | |
|---|---|---|---|
| Byte1 | Bit8-7 | | Keep 0 |
| | Bit6-1 | Zone index | Valid value 0 - 15(0x00 – 0x0F). |
| Byte2 | Bit8-6 | Zone setting value | 010: Value decrease (-1°C/-5%) 011: Value increase (+1°C/+5%) 100: Set open percentage 101: Set target setpoint Other: Keep setting value |
| | Bit5-4 | Control type | 01: Change type 10: Set to percentage control 11: Set to temperature control 00: Keep setting value(Must set 00 when no sensor) |
| | Bit3-1 | Power | 001: Change on/off state 010: Set to off 011: Set to on 101: Set to turbo Other: Keep power state |
| Byte3 | | Value to Set | When set percentage: 0-100 When set temperature: 0-250, setpoint=(value+100)/10 Other: Keep setting value |
| Byte 4 | | | Keep 0 |

Example:

Turn off the second zone:

    **0x55 0x55 0x55 0xAA   0x80 0xB0   0x0F   0xC0   0x00 0x0C**

          Header          Address   Id   Type   Length

    **0x20   0x00   0x00 0x00   0x00 0x04   0x00 0x01**

  Sub Type    normal length  repeat length repeat count

    **0x01 0x02 0xFF 0x00   0xF0 0xA1**

          Data           CRC

AirTouch will respond a message with sub type 0x21. (See next table)

## ii. Zone status (0x21)

Sending this message to AirTouch without any sub data (data length: 0x00 0x08, repeat count: 0x00, repeat length: 0x00) to request zone status from AirTouch.

**Note: AirTouch will send a zone status message automatically when zone status is changed.**

Data received from AirTouch:

No normal data (byte3 byte4: 0). **If the protocol is upgraded, this value may change. Use this specific value for data parsing.**

Repeat data count is the zone count in this message.

Each repeat data (8 bytes) contains one zone data. (byte5 byte6: 0x00 0x08). **If the protocol is upgraded, this value may change. Use this specific value for data parsing.**

| Byte in repeat data | | | |
|---|---|---|---|
| Byte1 | Bit8-7 | Zone power state | 00: Off<br>01: On<br>11: Turbo |
| | Bit6-1 | Zone index | 0-15 |
| Byte2 | Bit8 | Control method | 1: temperature control, 0: percentage control |
| | Bit7-1 | Open percentage | Current open percentage setting |
| Byte3 | | Set point | setpoint=(value+100)/10, 0xFF invalid |
| Byte4 | Bit8 | Sensor | 1: has sensor, 0: no sensor |
| | Bit7-1 | | *NOT USED* |
| Byte5 | Bit8-4 | | *NOT USED* |
| | Bit3-1 | Temperature | 0-2000: Temperature=(VALUE − 500)/10. |
| Byte6 | | | Other: Not available |
| Byte7 | Bit8-3 | | *NOT USED* |
| | Bit2 | Spill | 1: Spill active, 0: Spill inactive. |
| | Bit1 | Low battery | 1: Low battery, 0: Normal |
| Byte8 | | | *NOT USED* |

Example:

Request status of zones:

**0x55 0x55 0x55 0xAA　0x80 0xB0　0x01　0xC0　0x00 0x08**
　　　Header　　　　　Address　Id　Type　Length
**0x21　0x00　0x00 0x00　0x00 0x00　0x00 0x00　0xA4 0x31**
Sub type　　　　　　　　　　　　　　　　　CRC

AirTouch 5 response with data for 1 zone:

**0x55 0x55 0x55 0xAA 0xB0 0x80　0x01 0xC0 0x00 0x18 0x21 0x00 0x00 0x00　0x00 0x08　0x00 0x01 0x40 0x80 0x96 0x80 0x02 0xE7 0x00 0x00　0x01 0x64 0xFF 0x00 0x07 0xFF 0x00 0x00　CRCBytes**

Zone 1 data:

**0x40　　0x80　　0x96　　0x80　　0x02 0xE7　　　0x00 0x00**
**01000000　10000000　10010110　10000000 00000010 11100111** *000000*00 …

Power on, Temperature control,  Set point = (150[0x96]+100)/10 : 25,

Has sensor, Temperature = (743[0x2E7]-500)/10: 24.3

Zone 2 data:

**0x01        0x64        0xFF        0x00        0x07 0xFF              0x00    0x00**
**00**000001 **0**1100100 11111111 00000000 0000011111111111 …
Power off, Current open percentage setting: 100 (1100100 = 0x64) .
No sensor, invalid temperature

### iii. AC control (0x22)

AC control messages are to control all ACs. Each message to AirTouch is to control one or more specific ACs.

No normal data (byte3 byte4: 0).

Each repeat data (4 bytes) controls one specific zone. (byte7 byte8: 0x00 0x04).

| Byte in repeat data | | | |
|---|---|---|---|
| Byte1 | Bit8-5 | Power setting | 0001: Change on/off status<br>0010: Set to off<br>0011: Set to on<br>0100: Set to away<br>0101: Set to sleep<br>Other: Keep power setting |
| | Bit4-1 | AC index | Valid value 0 - 15. |
| Byte2 | Bit8-5 | AC mode | 0000: Set to auto<br>0001: Set to heat<br>0010: Set to dry<br>0011: Set to fan<br>0100: Set to cool<br>Other: Keep mode setting |
| | Bit4-1 | AC fan speed | 0000: Set to auto<br>0001: Set to quiet<br>0010: Set to low<br>0011: Set to medium<br>0100: Set to high<br>0101: Set to powerful<br>0110: Set to turbo<br>1000: Set to Intelligent Auto<br>Other: Keep fan speed setting |
| Byte3 | | Setpoint control | 0x40: Change setpoint.<br>0x00: Keep setpoint value<br>Other: Invalidate data. |
| Byte 4 | | Setpoint value | Available when byte3 is 0x40<br>Data to be sent = (setpoint * 10)-100 |

Example:

Turn off the second AC:

**0x55 0x55 0x55 0xAA   0x80 0xb0   0x01   0xC0   0x00 0x0C**
Header                Address      Id    Type    Length
**0x22   0x00   0x00 0x00      0x00 0x04      0x00 0x01**
Sub Type     normal length  repeat length  repeat count
**0x21 0xFF 0x00 0xFF   0xD3 0x47**
Data                CRC

Set the first AC to cool mode and second AC 26 degree:

**0x55 0x55 0x55 0xAA   0x80 0xb0   0x01   0xC0   0x00 0x10**

**0x22   0x00   0x00 0x00   0x00 0x04   0x00 0x02**

AirTouch will respond a message with sub type 0x23. (See next table)

### iv. AC status message (0x23)

Sending this message to AirTouch without any sub data (data length: 0x00 0x0A, repeat count: 0x00, repeat length: 0x00) to request AC status from AirTouch.

**Note: AirTouch will send an AC status message automatically when AC status is changed.**

Data received from AirTouch:

No normal data (byte3 byte4: 0). **If the protocol is upgraded, this value may change. Use this specific value for data parsing.**

Repeat data count is the AC count in this message.

Each repeat data (8/10 bytes) contains one AC data. (byte5 byte6: 0x00 0x08 or 0x00 0x0A, See 4.a). **If the protocol is upgraded, this value may change. Use this specific value for data parsing.**

| | | | |
|---|---|---|---|
| Byte1 | Bit8-5 | AC power state | 0000: Off<br>0001: On<br>0010: Away(Off)<br>0011: Away(On)<br>0101: Sleep<br>Other: Not available |
| | Bit4-1 | AC index | 0-15 |
| Byte2 | Bit8-5 | AC mode | 0000: auto<br>0001: heat<br>0010: dry<br>0011: fan<br>0100: cool<br>1000: auto heat<br>1001: auto cool<br>Other: Not available |
| | Bit4-1 | AC fan speed | 0000: auto<br>0001: quiet<br>0010: low<br>0011: med<br>0100: high<br>0101: powerful<br>0110: turbo<br>1001 - 1110:  Intelligent Auto<br>Other: Not available |
| Byte3 | | Setpoint | 0-250: Setpoint = (VALUE + 100)/10.<br>Other: Not available |
| Byte4 | Bit8-5 | | *NOT USED* |
| | Bit4 | Turbo | 1: Turbo active, 0: Turbo inactive |
| | Bit3 | Bypass | 1: Bypass active, 0: Bypass inactive |
| | Bit2 | Spill | 1: Spill active, 0: Spill inactive |
| | Bit1 | Timer status | 1: Timer set, 0: Timer not set |
| Byte5 | Bit8-4 | | *NOT USED* |
| | Bit3-1 | Temperature | 0-2000: Temperature=(VALUE − 500)/10. |
| Byte6 | | | Other: Not available |

| Byte7 | Error code | 0:No error |
|-------|------------|-----------|
| Byte8 | | Other: Error code of this AC |
| Byte9 | | *NOT USED* (Some version does not have those two |
| Byte10 | | bytes. Length defined in data Byte5-6, See 4.a) |

Example:

Request status of ACs:

**0x55 0x55 0x55 0xAA  0x80 0xB0  0x01  0xC0  0x00 0x08**
Header                Address    Id    Type  Length
**0x23  0x00  0x00 0x00  0x00 0x00  0x00 0x00  0x7D 0xB0**
Sub type                                        CRC

AirTouch 5 response with data for 2 ACs:

**0x55 0x55 0x55 0xAA  0xB0 0x80  0x01  0xC0  0x00 0x1C**
**0x23  0x00  0x00 0x00  0x00 0x0A  0x00 0x02**
**0x10 0x12 0x78 0xC0 0x02 0xDA 0x00 0x00 0x80 0x00**
**0x01 0x42 0x64 0xC0 0x02 0xE4 0x00 0x00 0x80 0x00  CRC Bytes**


AC 0 data:

**0x10     0x12     0x78     0xC0     0x02DA           0x0000        0x0000**
**00010000 00010010 01111000 ~~1100~~0000 ~~00000~~010 11011010  0000000000000000 …**
AC 0 is on, in heat mode and low fan speed and no error.
Current target setpoint setting: 22, VALUE=120(01111000 = 0x78), (120 + 100 )/10 = 22
Current Temperature: 23, VALUE=730(01011011010 = 0x2DA), (730-500)/10 = 23.


AC 1 data:

**0x01     0x42     0x64     0xC0     0x02E4           0x0000        0x0000**
**00000001 01000010 01100100 ~~1100~~0000 ~~00000~~010 11100100  0000000000000000 …**
AC 1 is off, in cool mode and low fan speed and no error.
Current target setpoint setting: 20, VALUE=100(01100100 = 0x64), (100+100)/10 = 20
Current Temperature: 24, VALUE=740(01011100100 = 0x2E4), (740-500)/10 = 24.

## b. Extended message(0x1F)

Extended messages are used to obtain the available modes, fan speeds, error codes of the ACs and names of zones.

When sending an extended message, **the address should be 0x90 0xb0**. When receiving the date for the extended message, **the last byte of address will be 0x9X**.

The first two bytes of the data are used to specify the specific command.

### i. AC ability (0xFF 0x11)

**This message is only available for console version 1.0.3 and above.**

Sending an extended message with data 0xFF 0x11 or (0xFF 0x11 [0-3]) to request the ability of all ACs or one specific AC.

Data received from AirTouch:

| | | | |
|---|---|---|---|
| Byte1 | | | Fixed 0xFF |
| Byte2 | | | Fixed 0x11 |
| Byte3 | | AC index | 0-15 |
| Byte4 | | Following data length | This data shows the count of following bytes belong to the ability of this AC.(24 at this moment) |
| Byte5-20 | | AC Name | 16 bytes in total. If less than 16 bytes, end with 0. |
| Byte21 | | Start zone index | Zone index start from 0 |
| Byte22 | | Zone count | |
| Byte 23 | Bit8-6 | | *NOT USED* |
| | Bit5 | Cool mode | 1: support, 0: not support |
| | Bit4 | Fan mode | 1: support, 0: not support |
| | Bit3 | Dry mode | 1: support, 0: not support |
| | Bit2 | Heat mode | 1: support, 0: not support |
| | Bit1 | Auto mode | 1: support, 0: not support |
| Byte 24 | Bit8 | Fan speed Intelligent Auto | 1: support, 0: not support |
| | Bit7 | Fan speed turbo | 1: support, 0: not support |
| | Bit6 | Fan speed powerful | 1: support, 0: not support |
| | Bit5 | Fan speed high | 1: support, 0: not support |
| | Bit4 | Fan speed medium | 1: support, 0: not support |
| | Bit3 | Fan speed low | 1: support, 0: not support |
| | Bit2 | Fan speed quiet | 1: support, 0: not support |
| | Bit1 | Fan speed auto | 1: support, 0: not support |
| Byte25 | | Min cool set point | |
| Byte26 | | Max cool set point | |
| Byte27 | | Min heat set point | |
| Byte28 | | Max heat set point | |

If there are more than one AC, the data will be repeated with relevant values. E.g. 2 ACs will receive 54(2+26+26) bytes data, 3 ACs will receive 80(2+26+26+26) bytes data.

It is recommended to request the AC ability when an AC back to normal state from any abnormal state.

Example:

Request ability of AC 0:

**0x55 0x55 0x55 0xAA   0x90 0xB0   0x01   0x1F   0x00 0x03   0xFF 0x11 0x00   0x09 0x83**

      Header          Address   Id   Type   Length      Data      CRC

AirTouch 5 response:

**0x55 0x55 0x55 0xAA   0xB0 0x90   0x01   0x1F   0x00 0x1A   0xFF 0x11 0x00 0x18**
**0x55 0x4E 0x49 0x54 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00**
**0x00 0x00 0x00 0x00 0x00 0x04 0x17 0x1D 0x10 0x1f 0x12 0x1f   CRCBytes**

AC 0 data:

**0x00 0x18 0x55 0x4E 0x49 0x54 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00**
**AC0   24   U   N   I   T**
**0x00 0x00 0x00 0x04   0x17      0x1D      0x10 0x1F 0x12 0x1F**
**      0   4   00010111 00011101   16   31   18   31**

Name of AC0 is "UNIT" and it has 4 zones, start with zone 0.

It has cool, heat, fan, auto modes and has low, mid, high, auto fan speeds.

Minimum setpoint for cool mode is 16, for heat mode is 18;

Maximum setpoint for cool and heat mode is 31.

## ii.   AC error information (0xFF 0x10)

**This message is only available for console version 1.0.3 and above.**

Sending an extended message with data 0xFF 0x10 [0-15] to request the error code of one specific AC.

Data received from AirTouch:

| | | |
|---|---|---|
| Byte1 | | Fixed 0xFF |
| Byte2 | | Fixed 0x10 |
| Byte3 | AC index | 0-15 |
| Byte4 | Error info length | Error info length(If no error, will be 0) |
| Byte5.. | Error info | String |

Example:

Request Error of AC 0:

**0x55 0x55 0x55 0xAA   0x90 0xB0   0x01   0x1F   0x00 0x03   0xFF 0x10 0x00   0x99 0x82**

      Header          Address   Id   Type   Length      Data      CRC

AirTouch 5 response:

**0x55 0x55 0x55 0xAA   0xB0 0x90   0x01   0x1F   0x00 0x1A   0xFF 0x10 0x00   0x08**
**0x45 0x52 0x3A 0x20 0x46 0x46 0x46 0x45   CRCBytes**

Data:

**0xff 0x10 0x00   0x08   0x45 0x52 0x3A 0x20 0x46 0x46 0x46 0x45**

### iii.   Zone name (0xFF 0x13)

**This message is only available for console version 1.0.3 and above.**

Sending an extended message with data 0xFF 0x13 [0-15] to request the name all zones or one specific zone.

Data received from AirTouch:

| | | |
|---|---|---|
| Byte1 | | Fixed 0xFF |
| Byte2 | | Fixed 0x13 |
| Byte3 | Zone index | 0-15 |
| Byte4 | Name length | |
| Byte5..n | Zone name | |

If there are more than one zone, the data will be repeated with relevant values.

Example:

Request name of zone 0:

**0x55 0x55 0x55 0xAA   0x90 0xB0   0x01   0x1F   0x00 0x03   0xFF 0x13 0x00   0x69 0x82**

Header           Address   Id   Type   Length       Data       CRC

AirTouch 5 response:

**0x55 0x55 0x55 0xAA   0xB0 0x90   0x01   0x1F   0x00 0x0A   0xFF 0x13**
**0x00           0x06           0x4C 0x69 0x76 0x69 0x6E 0x67   CRCByts**
**Zone0   Name has 6 chars   L   i   v   i   n   g**

Name of Zone 0 is "Living"

Request name of all zones:

**0x55 0x55 0x55 0xAA   0x90 0xB0   0x01   0x1F   0x00 0x02   0xFF 0x13   0x42 0xCD**

Header           Address   Id   Type   Length   Data       CRC

AirTouch 5 response:

**0x55 0x55 0x55 0xAA   0xb0 0x90   0x01   0x1F   0x00 0x1D   0xFF 0x13**
**0x00     0x06     0x4C 0x69 0x76 0x69 0x6E 0x67**
**Zone0     6     L     i     v     i     n     g**
**0x01     0x07     0x4B 0x69 0x74 0x63 0x68 0x65 0x6E**
**Zone1     7     K     i     t     c     h     e     n**
**0x02     0x07     0x42 0x65 0x64 0x72 0x6F 0x6F 0x6D   CRCByts**
**Zone2     7     B     e     d     r     o     o     m**

Name of zone0 is "Living".
Name of zone1 is "Kitchen".
Name of zone2 is "Bedroom".

### iv.   Console Version(0xFF 0x30)

Sending an extended message with data 0xFF 0x30 to request the version of the console.

Data received from AirTouch:

| | | |
|---|---|---|
| Byte1 | | Fixed 0xFF |
| Byte2 | | Fixed 0x30 |
| Byte3 | Update sign | 0-latest version, Other-new version available. |
| Byte4 | Version string length | |
| Byte5.. | Versions | Two consoles separated by ",". The first one is the master. |

If airtouch has two consoles, the first value is the version of the one that communicates with.

Example:

Request Versions:

**0x55 0x55 0x55 0xAA  0x90 0xB0  0x01  0x1F  0x00 0x02  0xFF 0x30  0x9B 0x8C**
       Header          Address    Id   Type   Length     Data       CRC

AirTouch 5 response:

**0x55 0x55 0x55 0xAA  0xB0 0x90  0x01  0x1F  0x00 0x0F  0xFF 0x30 0x00  0x0B**
**0x31 0x2E 0x30 0x2E 0x33 0x2C 0x31 0x2E 0x30 0x2E 0x33  CRC Bytes**

Data:

**0xFF 0x30 0x00   0x0B    0x31 0x2E 0x30 0x2E 0x33 0x2C 0x31 0x2E 0x30 0x2E 0x33**
          **Latest  Len:11   1   .   0   .   3    ,    1   .   0   .   3**